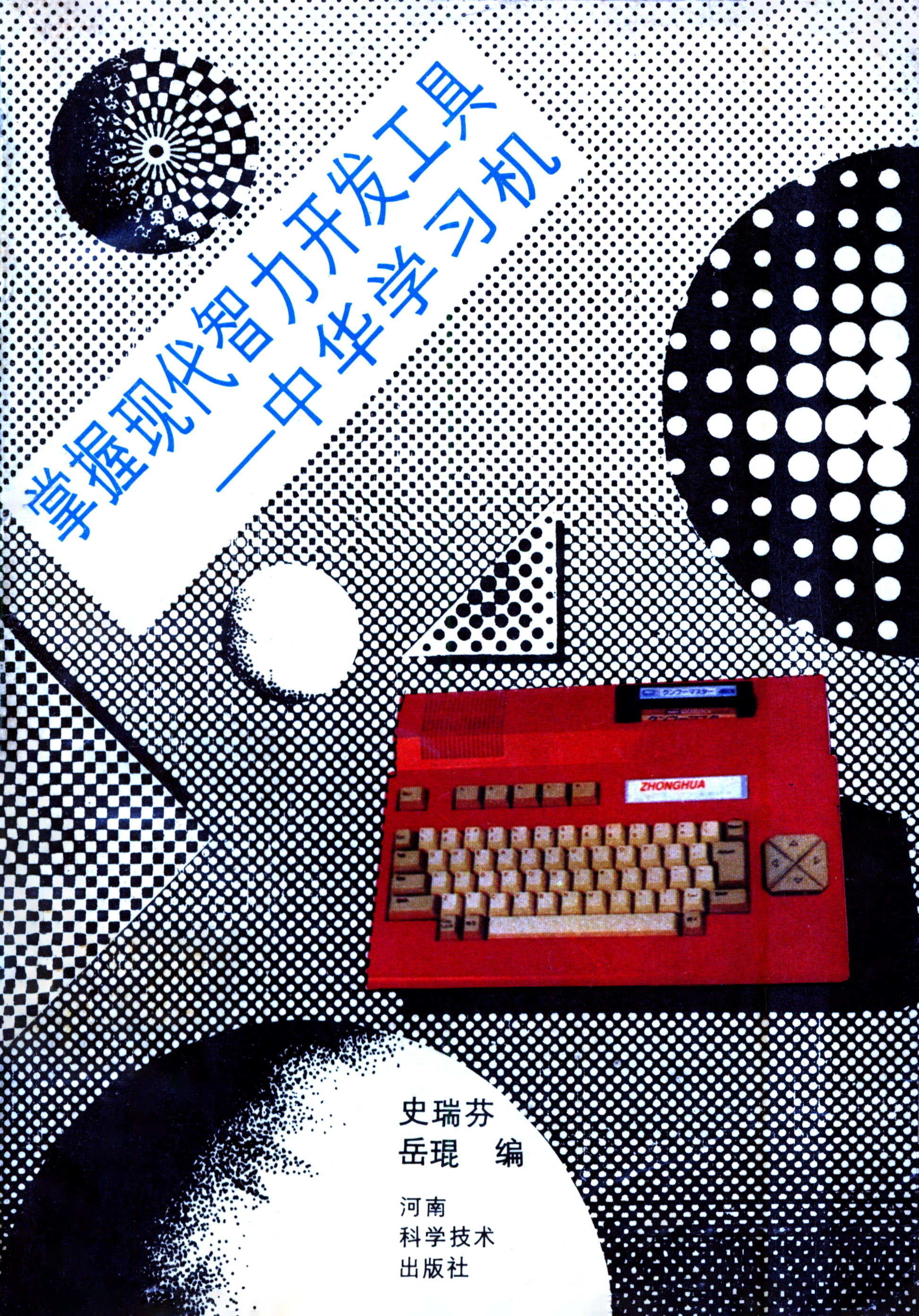

掌握现代智力开发工具
——中华学习机
ZHONGHUA
史瑞芬
岳琨　编
河南
科学技术
出版社

# 掌握现代智力开发工具

## ——中华学习机

史瑞芬　岳琨　编



河南科学技术出版社

## 内容提要

我国自行设计开发的中华学习机——适合中小学和家庭使用的计算机，将走进千家万户，它对开发青少年智力有重要意义。

为了配合中华学习机的普及，我们编写了这本书。本书根据不同层次读者的不同要求，分为五章讲了学习机的安装和使用，以及与此机配套使用的语言，即以简洁的语言使各类读者能取得满意的效果。

本书内容丰富、图文并茂，适合中等文化程度者阅读。

# 写在前面的话

当今时代知识更新加快，信息急骤变换，传统的教学方式面临着挑战，加速开发智力，快出人才已是当务之急，中华学习机由此应运而生。目前，全国已有三十余个厂家生产，近期它将以神奇的姿态“闯”入“寻常百姓家”。

如何更有效地利用中华学习机开发儿童智力，是值得研究和探讨的新课题。为此我们编写了《掌握现代智力开发工具——中华学习机》一书，献给广大的青少年朋友，献给为下一代日夜操劳的老师和爸爸妈妈们。

在编写本书的过程中，我们做了较深入的调查。智力开发是广泛的社会实践活动，由于各层次的读者对掌握中华学习机的要求不同，开发智力的着眼点不同，所以本书按不同层次读者的要求编排了学习内容，在开发智力方面提出了一些建议。

《一小时学会使用中华学习机》(第二章)，从内容讲是可以独立存在的，希望尽快掌握中华学习机、利用成品软件开发智力的读者，可以首先阅读第二章。工作繁忙的爸爸妈妈们，只要花上一小时的学习时间，就能与孩子一起享受电子游戏的乐趣；功课紧张的学生花上一小时的学习时间，就能请到一位耐心的“电子教师”。经过教学实践证明，中华学

习机的基本操作在一小时之内是完全可以掌握的，这也适应了时代的快节奏。

《驾驭海龟遨游神奇世界》（第三章），是为早期开发儿童智力而编写的。在老师和家长的帮助下，孩子很容易学会操纵海龟在屏幕上做各种饶有风趣的图画和游戏，甚至三四岁的孩子也能凭着他们的直觉和感受来使用学习机。从而培养孩子的想象力，发挥他们的创造性。爱因斯坦说：“想象力比知识更重要，因为知识是有限的，而想象力概括着世界上的一切，推动着进步，并且是知识进化的源泉。”成千上万的孩子从小就爱上了“电脑”，“电脑”启迪了他们的开拓精神与创造欲望。这里隐含了难以想象的潜力，意义该有多么深远！

《用智慧的钥匙打开知识宝库》（第四章），从解决实际问题入手，深入浅出地介绍了 BASIC 语言及编程技巧。BASIC 语言是众多程序设计语言中用得比较普遍的一种，适合初学者使用。它的特点：一是简单，使用范围广，大部分教学辅助软件都是利用BASIC 语言开发的；二是直观性强，具有计算机与人对话的功能，有问有答，使用方便。

我们学会用 BASIC 语言编写程序，不但可以解决我们身边的实际问题，而且可以从中尽兴地发挥个人的主观能动性和创造力。编写一个较为复杂的程序，一百个人如果分别独立进行程序设计，编出的程序虽然都能达到解决问题的目的，却可能没有某两个程序是完全相同的。这一方面体现了在程序设计中每个人的思路特点，即各自独特的灵活性和创

造性，它代表着个人的聪明才智，另一方面编制程序的本身就是进行着结构性思维和系统性思维的锻炼，从而能培养科学的思维方法。

《现代智力开发工具——中华学习机》（第一章），谈了关于使用中华学习机开发儿童智力的一些问题，其中介绍了造成儿童智力差异的因素，中华学习机在开发儿童智力方面所能起到的作用，以及如何更有效地开发儿童智力。作为父母都想使自己的孩子聪明能干、出类拔萃。本章论述的问题，希望对他们能有所启发。

《中华学习机使用指南》（第五章），较为系统地介绍了中华学习机及其使用方法，供读者查阅。

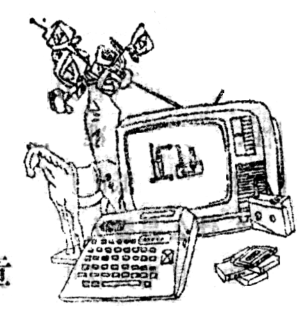
总之，本书是以不同层次读者的要求划分章节的，读者可以根据自己的需要选择有关章节进行学习。

在编写本书过程中，受到电子部、清华大学有关同志真诚的帮助，在此表示衷心的感谢。由于编写仓促，水平有限，不当之处恳请广大读者指正。

编　者

1988．3

# 目　录



# 第一章

# 现代智力开发工具

## ——中华学习机

## 一、儿童的智力需要开发

### 1.儿童智力存在着差异

我们经常可以看到一些爸爸、妈妈们在一起谈论孩子。他们有的为女儿学习成绩平平而发愁，有的为儿子不喜欢绘画、弹琴而遗憾，有的因孩子注意力不集中、学习劲头不大而大动肝火。

儿童智力差异到处可见。同是一个老师教的学生，一堂数学课讲下来，有的孩子作业十分八分钟就做完了，而有的则要花上一两个小时。儿童智力水平的高低是通过他们对事物进行分析与综合并做出相应的反应而表现出来的。有的孩子对所学知识领会得快、应用得活，善于举一反三、触类旁通，而有的孩子则反应比较迟钝，学习理解较慢，不能灵活

解决问题。

儿童在进行智力活动中所表现出的敏捷性、灵活性、广阔性、深刻性、准确性、系统性、创造性的差异，正是儿童智力水平的高低的反应。

**2.什么是智力，造成儿童智力差异的原因是什么**

智力是指人认识客观事物并用知识解决实际问题的能力。集中表现在反映客观事物深刻、正确、全面的程度上和应用知识解决实际问题的速度和质量上。对于儿童的智力结构，一般认为应包括：观察力、记忆力、想象力、注意力、思维能力和操作能力。

造成儿童智力差异的原因是什么呢？一种观点认为智力差异是天生的，即遗传决定论；另一种观点认为差异由环境决定，即只要对影响儿童所处的环境条件加以控制，就可以把儿童培养成某种类型的人，完全否定遗传的作用；还有一种观点认为，儿童智力的发展是由遗传和环境两种因素所决定的，各占一半，儿童智力的发展是被动的，由环境与教育在遗传素质上施加影响的结果。

以上观点都有片面性，我们认为遗传因素是儿童智力发展的生理前提，社会历史条件是智力发展的背景，教育和环境是智力发展的外部条件，儿童个体的实践活动是智力发展的基础，儿童心理的内部矛盾是智力发展的动力。

首先，我们承认智力发展与遗传禀赋有关，健全的大脑是儿童智力发展的前提。从生理解剖观点看，一般正常的儿童虽然无遗传缺陷，但由于每个儿童遗传的基因不同，则他们

的大脑和感觉器官的生理解剖特点不同。比如：有的儿童细胞核内少一条X染色体，结果会造成这个孩子语言能力很强，而空间知觉较差。所以说，遗传因素是形成智力差别的生理前提。但是如果说遗传决定智力水平的高低，则是错误的，因为另一方面应看到环境和教育是智力发展的外部条件；幼苗是否能长成参天大树，那要看它的生活条件；从玉石到精美的工艺品要靠独具匠心地雕刻。莫扎特3岁就能演奏钢琴，约翰·斯图尔特弥尔3岁就能看懂拉丁文的古典著作，全靠他们的父亲所进行的严格而有计划的教育。

心理学家斯基尔斯曾对环境的影响进行了专门的对比研究。他把孤儿院2岁的婴儿分成两组：一组为实验组，由智力比较差的13名儿童（平均智商为63.4）组成，送至有社交刺激的环境中培养；另一组由比较聪明的12名儿童（平均智商为86.7）组成，留在孤儿院缺乏社交刺激的环境中生活。一年半后发现，原来较笨的儿童智商提高了（平均91.8），而原来较聪明的儿童的智商反而下降了（平均60.5），到成年以后，原来较笨的儿童在学习与工作方面也比原来聪明的儿童强得多。（引自〔美〕克雷奇等著《心理学纲要》）

这说明，即使具有较好的先天素质，离开了环境与教育的影响也不能成才。

儿童智力的发展不是被动地接受外界条件的影响，而是通过儿童个体实践活动得以发展的。恩格斯指出，“人的智力是按照人如何学会改变自然界而发展的”。从生理学观点看，人生下来脑细胞几乎没有功能，随着外界的刺激，时间

的推移，智慧的增长，分散的脑细胞才集结起来，相互联系，脑功能逐步健全。同样，儿童的观察力、记忆力、想象力、思维力等都不是通过家长、教师单方面的教授所能发展的，而是通过儿童自己独立的活动才得到发展的。

另外，儿童智力的发展是靠内外因的相互作用、相互联系、相互影响取得的。比如，孩子上课学习了四则运算，而原先对此并没认识，这就需要通过课堂教（外因）、个人理解和思维（内因）来认识，再动手运算，加深体会直到逐步掌握四则运算；这个从不会到会就是依赖于遗传的因素、环境与教育的条件，同时又取决于儿童心理的内部矛盾；内因由外因的转化而提高智力。所以说儿童智力发展的效果取决于儿童个人的实践和主观能动性发挥的程度。我们的一些爸爸妈妈们对孩子操心过度，包办代替太多，这种自以为对儿童教育有利的做法，实际是无意地影响和限制了儿童智力的发展，扼杀了孩子的创造性和主观能动性。

中华学习机可以作为你教育孩子的得力助手。

## 二、谈谈中华学习机

### 3.教育改革的大趋势

近年来，新技术飞速发展，知识更新空前加快，据统计，近30年人们创造的科学技术知识超过了过去几十万年间人类知识的总和。人们面临着挑战：不断更新的知识、无限

的书籍和报刊等资料与人们有限的阅读时间之间的矛盾；按几何级数膨胀的庞杂纷云的瞬息万变的信息流，与人们的理解能力、控制能力和应用能力的矛盾。如果一成不变地按照固定的传统方式教与学，是永远也赶不上急骤变化的知识流与信息流的。传统的教育形式、教育方法、教育内容以及教学管理都要进行改革，计算机与基础教育、家庭教育相结合，是教育改革的大趋势。当前，世界上许多国家的计算机教育重点以惊人的速度进入中学、小学和家庭。普及型计算机（学习机）已形成系列。在日本，目前平均每6户一台学习机；美国1984年销售家庭用学习机450万台，1986年销售380万台，平均每6个人一台学习机。

从另一个角度讲，"计算机科学将继自然语言、数学之后，成为第三位对人的终生都有大用的"通用智力工具"，这是20年前美国著名计算机科学家G·伏特斯的预言，今天已被证实，将来必定进一步被证实。

外国人往往欣赏中国人的聪明才智。古老文明的中华民族，有着丰富的智力资源和脑力劳动优势，但是我们是否考虑过，社会发展到今天，如果不同时有效地利用"电脑"，这个人脑的优势就会丧失，或者说不懂得利用电脑的人脑，会落后于用电脑作为辅助的工具的"人脑"的。

当前，我们不少人已认识到计算机作为通用智力工具的

战略意义。如果我们在智力开发上不充分利用计算机，将会与经济发达的国家造成更大的差距。在今后十年或十五年内走上工作岗位的亿万中、小学生必须迅速掌握信息时代的工具——计算机，并在学习计算机的过程中开发智力。这也是迫在眉睫的问题。

### 4.什么是中华学习机

中华学习机是我国自行开发的，适合中学、小学及家庭进行智力开发、辅助教学、事务管理的微型计算机系列，它与 APPEL-Ⅱ（苹果机）兼容，是由国家科委、国家教委、电子部等单位联合组织开发的。现在我国已有30多个厂家生产，投放市场的有 CEC-Ⅰ中华学习机，CEC-Ⅱ 型中华学习机正在开发，“小蜜蜂”中华学习机已经过鉴定。

中华学习机适合中国的国情，它性能价格比高，汉字功能强，在目前大多数家庭都有电视机和收录机的情况下，只要再买一台中华学习机作为主机，以电视机作为显示器，用一般家用收录机作为外存贮器，就可以组成微机系统，再选用盒式录音带录制的教学软件和游戏软件，便可以在家里上机了。

### 5.中华学习机能帮我们做些什么

(1) 中华学习机——幼儿的好朋友

现在年轻的爸爸妈妈们都感到独生子女孤单得很，想给孩子找个好朋友，中华学习机是可以胜任的。

中华学习机可称为万能游戏机，目前的不少游戏软件都可以在中华学习机上运行，比如《空间大战》、《酒吧》、《海战》等。在游戏中总有一个对手和孩子比赛（学习机可充当对手），比赛过程中可以使孩子感受到胜利的喜悦和失败的苦恼。从而锻炼了观察能力、反应能力和稳定而持久的注意力。变换莫测的电子游戏将给孩子带来极大的乐趣，扩大了孩子的视野。



中华学习机以它鲜艳的色彩、变化的图像、模拟的声响，让孩子在玩的过程中受到视觉和听觉的锻炼，同时还学会认识动物，分辨颜色、形状，数数和比较数的大小。当孩子回答对时，屏幕上就可以显示分数和赞美的动画，还可以配上一段音乐给以鼓励。

在操作中，孩子需要学习使用键盘。对幼儿进行指法训练能锻炼他们手脑并用，同时也从小教会他们打字的技能。

当孩子们坐在荧光屏前用LOGO语言使海龟作图时，屏幕上将随着孩子的操作而出现千奇百怪的图形，这时孩子将展开想象的翅膀在幻想中翱翔。

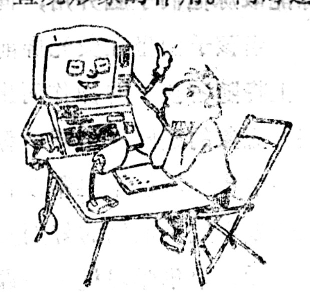
中华学习机不但帮助爸爸妈妈们启迪了孩子们的智慧，也一定会成为他们的好朋友。

（2）中华学习机——课堂教学的助手

中华学习机可以以它直观而形象的画面模拟一些难以实际操作的实验过程和难以表达的动态的、立体的图形。比如数学中各种函数的图像，立体几何中的复杂图形，都可以利用中华学习机所具有的动态图形方式去表现函数的轨迹及立体图形的旋转。这有助于老师的讲述和学生的理解使他们在轻松愉快的气氛中完成教学任务。

物理实验是物理教学的重要内容。目前由于不少中学受实验条件的限制，使得许多实验无法进行。中华学习机可以充当实验室，模拟物理实验，图文并茂，动画表演，形象直观。比如物体沉浮条件实验，学生们坐在中华学习机显示器屏幕前，便可以看到物体在各种条件下沉浮的情况。并且从中华学习机的扬声器中传出的娓娓动听的讲解，使学生处于视觉和听觉的双重刺激之中，从而加深对此概念的理解。

在化学学习中，中华学习机更可以大显身手了，如它可以用来解决某些微观现象变化的过程，显示S亚层电子云和P亚层电子云形成的过程，在空间的形状、伸展方向、密度等等，对电子云概念的形成起到了直观形象的作用。同时还可以用来演示原子核外电子运动状态，从而解决化学中这一难点。



当然，上面列举的仅仅是中华学习机用于辅助教学的极少一部分，但足以说明中华学

习机——课堂教学的助手。

（3）中华学习机——耐心的家庭教师

在课外，中华学习机可以充当家庭教师，去辅导、督促孩子们学习。

比如：中华学习机能帮助小学生复习巩固汉语拼音的拼读方法，还能帮助他们进行口算练习，讲解各类应用题。

中华学习机能给孩子们出题考试，完成后当场评分，并显示答案。

中华学习机除了可以辅导中学生复习英语单词、解数学习题，提出有关元素周期表的问题外，还可以回答学生的疑难问题。当然，有的问题它会给出正确的回答，有的问题或许学习机一时答不出来（软件不可能面面具到），这时，你便可以从书本上查出正确答案告诉中华学习机，这样，它就学会了，下次再问到这个问题时，它就能回答出来了。

中华学习机对高考的学生来说帮助会更大，它可以将大量的历届高考试题、练习题存贮起来，当考生复习到某门功课、某一部分时，学习机会自动地列出相应的试题，供学生练习，并能批改和给出标准答案。不论你什么时候需要它的帮助，不论你需要重复多少遍，它总是不厌其烦——真能称得上耐心的家庭教师。

（4）中华学习机——软硬件的开发工具

中华学习机还可以在企业决策、方案预算、工业控制、自动化、财会管理以及人工智能等方面广泛应用。

中华学习机还能作为软硬件开发的工具在更加广泛的领

域里发挥作用。

(5) 中华学习机——认真负责的管家

中华学习机可以帮你建立家庭档案，记下值得纪念和庆贺的日子，如家庭成员和亲友的生日；还可以承担家庭财务管理，记下存款的数目、日期、号码及物品的存放地点。

中华学习机还可以帮助你编写资料，比如你零乱的、随机输入的资料，它可以帮你归纳整理，以至打印在纸上。如果你要查找某些资料，它便可以迅速完成，做到准确无误。

随着家庭现代化程度的提高，家用电器逐步增多，如何将它们管理起来从而更有效地发挥作用呢？在这方面学习机堪称行家里手，将家庭中24小时之内需要做的工作编成程序送入学习机（当然，除中华学习机主机外，还要有相应的接口电路、转换器等）这样，各种家用电器便可在中华学习机的控制下按程序工作了。



(6) 中华学习机——家庭娱乐的多面手

人们在物质生活水平提高的基础上，对精神生活的需要也日益增强，学习机可以作为电子游戏机供全家人娱乐，还能陪着你下象棋、围棋。

中华学习机设有发声装置，通过程序控制可以发出各种奇妙的声音，可以谱写乐曲，还可以当作电子琴进行弹奏。

总之，中华学习机在学校、家庭和其它方面有广泛的应用价值。随着对中华学习机系列软件的研制和开发，用途将会更加广泛，以至会起到我们意想不到的作用。

### 6.中华学习机是如何工作的

从上述事例可知，中华学习机可以为我们做许许多多的事情，真有些“神奇的魔力”了。那么，它的魔力来自何方呢？今天就让我们揭开这个秘密，看一看它是如何工作的。

我们首先来分析一下人脑是如何处理问题和解决问题的。

人的解题过程大致如下：通过眼和耳朵等感觉器官将看到的和听到的问题送入大脑（输入）。大脑的中枢神经经过分析，指挥手、嘴等器官或者运用工具去执行、去计算（控制、运算）。然后将执行或运算的结果通过嘴说、手记等方式表现出来（输出）。

而中华学习机作为“电脑”的一种，它又是如何摸仿人脑进行工作呢？这就要看组成电脑的各部分结构以及它们之间是如何协调工作的。

中华学习机大致分为以下几个部分：

(1) 输入设备

它就像人的眼和耳，负责接收信息，并且转换为计算

机可以存放的形式，存入内存贮器保存。因此，可以说输入设备是计算机与外界进行联系的纽带。

中华学习机的主要输入设备是键盘，它安装在主机外壳上，共有69个键，包括标准打字机键盘的各种键，并且按照标准打字机的键位排列。键盘的作用是通过操作者按键盘上的键，便可以把信号送入计算机。所以掌握好键盘操作，便可以迅速地将信息送入计算机。

（2）内存贮器

存贮器是计算机的记忆装置，它的存在，使计算机有了记忆功能。

存贮器又分为内存贮器和外存贮器。内存贮器安装在主机箱内，它是由半导体元件组成的，读写速度快。

内存贮器像一个巨大的"仓库"，每一层都排列着很多"盒子"，每个盒子都编上号码，这就是存贮单元的"地址"。"盒子"内存放若干位二进制信息，每个"盒子"里的内容组成一个字。一个字所包含的二进制位数就称为"字长"。计算机的字长越长，计算的精确度就越高。中华学习机的字长为八位，这就是说，一个"盒子"，即一个存贮单元存放八位二进制代码；一般习惯称八位二进制代码为一个字节。因此，中华学习机的一个存贮单元存放一个字节的二进制代码。

存贮单元的多少表示了存贮器容量的大小。存贮器的容量以K为单位，$1K=2^{10}=1024$ 个存贮单元。中华学习机的随机存贮器有64K，即65536 个存贮单元。

内存贮器又分为随机存贮器（RAM）、只读存贮器（ROM）和可编程只读存贮器（EPROM）三种。

随机存贮器（RAM）又称为读写存贮器，可以随时写入或读出。但一旦切断电源，信息便会全部消失。

只读存贮器（(ROM）是一种一旦“编入程序”之后，就只能进行读操作的存贮器，但当电源被切断之后，信息仍能保留。所以常用来存放不变的程序和常数。中华学习机有32K的ROM，上面固化着监控程序和BASIC语言。

可编程的只读存贮器（EPROM），故名思意，存贮的内容可由用户抹去，但要再次写入必须使用电脉冲。

当键盘接受到信息之后，先进行转换，使之成为计算机可以接收的信息，然后如同大脑记忆一样，将此信息记入内存贮器，暂时保存下来。

（3）控制器

控制器就如同人的大脑中枢神经一样，指挥着各部分协调工作。

前面讲到键盘接受到信息之后存放在内存贮器里。信息可以分为两部分：一部分为数据，另一部分为命令或程序指令。指令来自于人们编制的程序。计算机在工作时，控制器就一条一条地将命令或指令取出，分析各条指令（命令）是要进行什么操作，操作的数据存放在内存哪一个单元，然后指挥各部分去工作。

（4）运算器

运算器是在控制器的控制下完成各种算术运算和逻辑运

算的部件。在运算过程中，不断地从存贮器取出数据，并将计算结果送入内存贮器保存起来。

人们把控制器，内存贮器和运算器合称为中央处理机或主机，简称 CPU（即英语 CENTRAR PROCESSING UNIT 的字头）。它是计算机的核心，它使计算机有了分析问题和处理问题的能力，即成为“电脑”。中华学习机的中央处理机CPU 是 6502，它是一个八位处理器。时钟频率为1MHz，数据总线八条，地址总线16条，寻址空间64K字节，它安装在主机线路板上。除此之外，主机线路板上还装有64K 的随机存贮器、32K 的只读存贮器、存贮管理部件MMU、输入/输出管理部件IOU，可编程阵列部件 PAL。另外还留有扩充汉字系统的插座。若需要增加汉字系统只要插上一片固化有汉字系统的ROM片子27256，两片固化有汉字字库的ROM片子和两片TTL电路即可。

（5）外存贮器

内存贮器虽然具有读写速度快的特点，但是由于它的存贮容量有限，并且信息不能长期保存，因此，又设置了外存贮器，外存贮器都是利用磁性材料的剩磁效应——靠磁头把需要的信息存入或读出。

中华学习机的外存贮器采用磁带机（可以用录音机来代替）和软磁盘存放暂时用不到的信息，同时也可以随时同内存贮器交换信息。例如将录音机连在主机上，便可以很方便地把信息从内存贮器存入磁带或者从磁带读入内存。

软磁盘机存贮信息比录音机速度快，容量大，而且动作可

靠。它由磁盘驱动接口、磁盘机、磁盘三部分组成。磁盘驱动接口卡是主机和磁盘之间传送信息的桥梁，这部分已做在主机板上。磁盘机本身与唱机相似，由一个电机带动磁盘转动，机上有一个磁头可读取磁盘上的信息或者将信息写到磁盘上。

外存贮器如同我们在记忆时需要将一些信息记在纸上或本子上一样，一旦忘记，便可以从外存贮器（本子）上获取。

(6) 输出设备

计算机在运行中出现了问题该如何显示呢？或者计算的结果如何表示出来呢？这需要有输出设备。目前微型计算机主要的输出设备是显示器和打印机，显示器可以随时将我们输入的命令、程序显示出来，也可以将计算结果显示出来，而打印机则是将这些信息打印在纸上。中华学习机的显示器可以用黑白电视机或者彩色电视机来代替，而打印机多采用点阵式的打印机，如MX-80、FX-80、MX-100、FX-100等型号打字机。

通过以上的介绍，大概对电脑是如何工作的过程有了一些了解，它的结构模拟了我们人脑，因而能分析、处理问题。

总之，中华学习机是这样工作的：

①从键盘（输入设备）接受信息（键入的程序、指令或数据），也可以从磁盘或者磁带上（外存贮器）直接接受信息（事先已经存贮在上面的程序或数据）。

②接受到的信息可以直接显示在屏幕上供人们了解或修改，同时转换成计算机可以接受的信息存贮在内存贮器里。

③控制器将接收到的程序或指令从内存贮器里取出并进行分析，然后协调中华学习机的各个部分工作。同时，根据程序中指出的操作数地址，将操作数取出送入运算器。

④运算器在控制器控制下完成各种运算。

⑤运算的结果送显示器和打印机（输出设备），必要时送磁盘或磁带（外存贮器）保存起来。

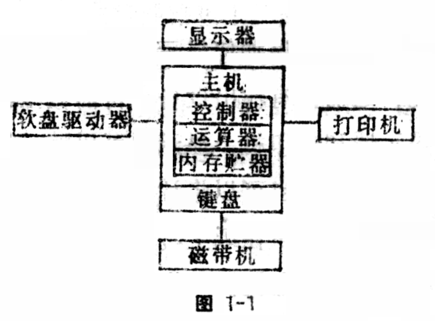



图 1-1

图1—1是中华学习机系统组成图。

除此之外，中华学习机还具有汉字输入、输出功能，这样一来就给使用者带来极大的方便，中华学习机的汉字系统是采用固化的 ROM，因此可以节省内存空间，而且实现起来也很方便。

中华学习机和苹果机（APPLE-Ⅱ机）兼容。它的功能相当于一台苹果机，而且有所增强，能够在苹果机上运行的各种软件在中华学习机上都能运行，这样一来，就为中华学习机的推广提供了极大的方便。

# 三、中华学习机为什么能有助于开发儿童智力

前面提到智力结构包括感知力、注意力、记忆力、思维能力、想象能力以及操作能力等，并且还提到智力开发的关键是儿童个人的实践和他的主观能动性发挥的程度。在这些方面，中华学习机能为开发儿童的智力大显身手。

## 7.中华学习机能使儿童提高感知力

什么是感知力呢？

感知力是指客观事物作用于人的感觉器官后，而在头脑中产生的反映能力。感知是认识客观事物的开端。

感知力包括视觉、听觉、空间知觉和时间知觉，人们的感知力是在实践活动中发展起来的，它是反映人们智力高低的显著标志。

中华学习机以它直观形象的画面和模拟音响来刺激儿童的多种感官，提高视觉能力（色彩、色度和形状的辨别）和听觉能力（音调的高低）。

中华学习机在运行应用程序或者游戏程序时，速度是非常快的。这就激发儿童在人—机对话中要反应敏捷，迅速作出回答。游戏程序更能刺激儿童的感官。它一般都以紧张的节奏，瞬息变化的画面出现。如在游戏“星球大战”中，“太空人”一个劲地扔炸弹，儿童操纵着的飞机一方面要

想办法“躲避”炸弹，一方面要有效地还击，从而锻炼了孩子的反应能力。低幼儿童观察感知事物时，往往比较笼统、模糊，又易被新颖的、非重点的方面所吸引，离开了观察目的；通过操作中华学习机能够逐步纠正上述这些毛病。

## 8.中华学习机能使儿童提高注意力

注意，就是心理活动对于一定事物的指向和集中，它是保证人们心理活动清晰，全面地反映事物，提高工作的效率和正确性的必要条件。

注意力稳定性（也就是持久性）的培养对幼儿来说是非常重要的，一些学习成绩不理想的学生，往往注意力不够集中，上课心不在焉，而注意力强的孩子能在相当长的时间内连续保持注意力。一般情况下，七岁的儿童可以连续注意10～15分钟，十一二岁的孩子能保持注意力30～40分钟。

中华学习机以它动态的画面能够引起儿童的注意，容易引起他们的兴趣；在学习和娱乐中不断强化注意力。

另外，中华学习机可以采取多种方式，灵活地讲授知识，明白易懂，生动有趣，孩子们兴趣提高了，注意力也就集中了。

## 9.中华学习机能使儿童提高记忆力

记忆，是过去的感知和经验在人脑中的反映。人类如果没有记忆，将永远处于新生儿状态。

中华学习机可以帮助儿童在形象记忆的基础上加强和提

高抽象记忆。

比如，小学数学中的行程问题：速度＝距离÷时间，若只进行抽象讲解，孩子们不容易记住这个公式，中华学习机可以图形动态演示：屏幕上可以在不同距离和不同时间条件下，出现不同速度的小汽车，孩子们逐步加深对行程问题的理解，从而通过形象记忆过渡到抽象记忆。中华学习机很方便地就为儿童创造了这个记忆的条件。

记忆的过程可分为：识记——再识和回忆——保持，记忆是个复杂的心理过程，如何保持记忆呢？需要及时复习，反复回忆，采用多样化的复习方式。中华学习机恰恰可以给我们提供这样的方便。只要将软件设计成能够反复多样地提出问题，便可帮助孩子们进行复习，保持记忆。

### 10.中华学习机能使儿童提高想象力

想象是幼儿创造能力的起点。文艺复兴时期的伟大画家、科学家达·芬奇说过，他在幼年时总觉得家里墙壁上的污迹和裂纹就象是魔女在飞舞，就象是奇形怪状的动物在格斗一样。

列宁指出：幻想是可贵的品质，积极的幻想不仅是儿童学习的动力，而且对于他们精神面貌和道德品质的形成均起着重要作用。

中华学习机的LOGO语言（以后章节要讲到）就能帮助孩子们去提高想象力。LOGO语言绘图功能很强，通过简单的语法学习，孩子们便可以操纵一个称为“海龟”的作图装置在

屏幕上绘制、拼装和组成各种图形，使图形旋转变换，甚至可以使图形运动；任何一种新的想法，几乎都能产生一种奇妙的图形，这样，孩子的想象力和创造欲便会油然而生。

### 11.中华学习机能使儿童提高思维能力

思维是通过对事物分析、比较、抽象、综合与概括的过程，是认识事物的本质和规律的活动。

要培养孩子的思维能力，首先要创造出使他积极思维的环境，提出他们感兴趣的问题，激发起解决问题的愿望，从而使孩子开扩思路，促使思维的发展。中华学习机能够提供这种方法和手段。人—机对话的方式促使孩子经常处于积极思维状态。

从孩子思维发展过程来看，一般 3 ～ 7 岁的孩子处于形象思维阶段，他们往往借助于事物的具体形象进行思维。小学阶段的儿童，处于从形象思维向抽象思维发展的转变时期，随着年龄、见识的增长，逐步学会通过抽象的概念，进行推理、判断。到了青年期往往抽象思维便占据了主导地位。

儿童开始接受的概念都是具体的概念。比如一只小狗、三只小兔等，逐步过渡到能够概括出这些具体的实物有些什么特性，从而进行分类、区别。如小狗和小兔都是动物，他们的不同点是耳朵不一样长等等。通过屏幕上画面的比较逐步了解事物的本质和特征。

儿童开始接受的概念是孤立的，而中华学习机可以培养孩子们分析和了解事物之间的联系。如中华学习机的LOGO

语言，就可以促使孩子将事物联系起来考虑，即孩子可以先在屏幕上画一个正方形，然后再画一个三角形；这两个图形是分离的。如果把三角形放在正方形上面不就变成一座小房子了吗(图1—2)？如果再复杂一些，画两个大小不同的长方形和两个圆，看孩子们能不能组成一辆汽车（图1—3）。开始孩子可能装得不对，但在家长的鼓励启发下，一定能完成这一组合。

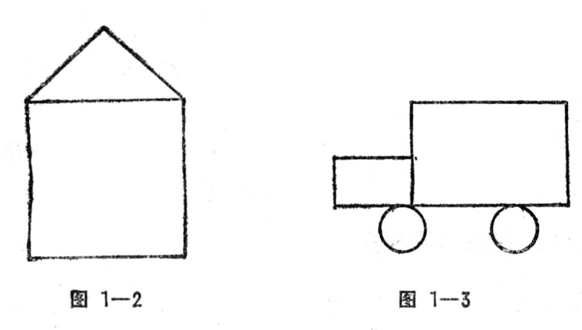


图 1—2　　图 1—3

在开发儿童思维能力时要注意培养结构思维观念，发展新的思想方法。结构是指某一系统的各个要素的结合方式或组合方式，任何系统都是按一定结构组成的整体。

结构化思维是这样一种过程，把比较复杂的问题分解成若干较小的和容易解决的问题来解决，然后在此基础上综合得出原来复杂问题的解决方法。

系统化思维观点认为一切物体都是一个整体系统，现实物质世界的一切事物都是以系统方式存在发展的，并且每个系统都与环境不断进行物质与能量交换，即对外开放型；系统内部是按层次等级有机联系起来的，即对内层次型。

这两种新型的思维方式，从幼儿时期家长就要有意识地进行培养。而利用中华学习机的LOGO语言就可以培养孩子们的结构化思维方式。

对于系统化思维方式的培养是需要多方面的实践和锻炼。程序设计本身就是一个理想的实践过程。要想设计出一个完美的程序，必须首先整体构思，系统地考虑设计对象的物理模式，决定程序的结构和流程——这些都要树立系统的观点。作为系统内部，又要分成不同的层次；程序处理起来也必须按层次做出相应的区分。因此，通过对程序结构的安排和对纵横交错的子程序的分析，会使孩子受到层次化思维的训练。

### 12.中华学习机能使儿童提高操作能力

俗话说“心灵手巧”。看来心灵和手巧密切相关。儿童的操作能力对于智力的发展有很大的促进作用。使用中华学习机离不开动手操作，特别是玩电子游戏时，要想取得胜利，必须按键时机准确、迅速、果断，这时必需手脑并用，思维和动作协调。实践证明，经过一段时间的训练，孩子的反应能力会增强。

中华学习机键盘是按照标准英文打字机键盘的键位排列的，从小掌握正确的键盘操作要领，锻炼打字手法，对提高打字速度是非常必要的。现在的小学生，等他们长大时，我国的计算机使用已经相当普及，每个人都有相当多的工作要在键盘上做，学好中华学习机的键盘操作，除了有利于智力

发展之外，还掌握了打字这门技能。

## 四、怎样利用中华学习机更有效地开发儿童智力

### 13.根据不同孩子的特点确定学习内容及选择合适的软件

儿童智力的发展有明显的差别，即有的智力显露较早，而有的儿童发育较晚。莫扎特在他父亲的指导下3岁就写出了《小步舞曲》。而爱因斯坦4岁才会说话，7岁才会念书，但他们都成为出类拔萃的人物。关键是要因才施教。所以，在本书的学习过程中，针对不同儿童的智力水平和特点，要选学相应的章节。一般来说，年龄较小的孩子先学“海龟作图”，而青少年可以学一些程序设计；对于学习较为吃力、时间较紧迫的学生，可以直接学会使用教学辅助软件，这对学习成绩的提高有一定帮助。这只是对一般情况而言，老师和家长可以根据孩子的情况安排学习计划。

儿童智力的差异不仅表现在智力水平的高低不同，而且表现为智力活动的类型不同。如有的儿童思维中形象因素占优势，有的儿童抽象的概念和逻辑联系占优势；前者属于形象思维型，而后者属于逻辑思维型，对于前者和后者采用的学习内容应有所区别。

要充分发挥中华学习机的功能，必须有相应的软件来支持。开发儿童智力更要注意选择好合适的软件，如果选择不当

会影响孩子的兴趣，起不到开发智力的作用。

选购的软件一定要能在中华学习机上运行；软件的难易程度要适合孩子的年龄和智力水平；教学软件的内容应符合孩子使用的教课书，等等。

目前，我国开发儿童智力方面的软件还不丰富，为了适合自己孩子的特点，老师和家长需要亲自编写一些程序，这样效果往往更佳。比如，我们可以用LOGO语言编一些适合三四岁孩子使用的程序，让他们可以凭借自己的直觉和感受来操作画图（参考“第三章　六”）。还可以用 BASIC 语言编写程序，让孩子和学习机对话，当屏幕上出现孩子名字时，他一定会倍感亲切，增加学习欲望（参看“第四章　一”）。

### 14.兴趣——欲望——持之以恒

最近,碰到这样一件事:一对年轻父母省吃俭用给孩子买了架钢琴，并高薪聘请了教师,但孩子哭着闹着不愿意学,每次老师来讲课,都要靠父母软硬兼施才能坐到琴凳上,每天练琴也要靠父母强迫，有一天,趁家人不注意,孩子用菜刀砍坏了琴键。这是为什么呢？因为孩子产生了逆反心理。由此可以看出，在教与学这两方面，首先学生要有兴趣、有要求、有欲望。我国古代教育家孔子就强调调动学生的积极性，他说“不愤不启，不悱不发”，意思是讲授要掌握时机，等学生“愤”“悱”之时，即有迫切的求知欲望时，才能施教。

如果孩子不愿意学怎么办呢？

日本铃木镇一先生创办了“小提琴教室”，入学学拉小提琴的是两三岁的孩子，这些孩子有的也是被母亲硬拉来学琴的，他们对小提琴并不感兴趣，来到之后，东张西望，甚至到处乱跑，铃木先生并不急于教他们拉小提琴，故意不给孩子提琴，而是先教母亲拉小提琴，以此来激发他们的兴趣，后来，孩子们对琴产生了兴趣，一动不动地坐在那里看那些和自己差不多大小的小朋友练琴，再过两三个月他们也听会了一些曲子，非常想亲手拉一拉小提琴，这时，铃木先生还不发给他们提琴，直到他们实在坐不住了，才让他们参加训练。这时会看到，一旦孩子产生了浓厚兴趣，进步是很快的。所以说兴趣是精通的基础。

我们的家长、老师在和孩子一起学习中华学习机的时候，首先要让孩子对学习机感兴趣。比如让他们看看中华学习机在运行一些有趣的软件时，产生的画面和特殊效果等，从而激发起他们的学习欲望。

在学习中华学习机的过程中，从某种意义上讲老师和家长的主要任务是激发孩子的学习兴趣与欲望。即要创造条件，不断提出新的问题，调动孩子积极主动去学，从而达到持久以恒。

### 15.靠孩子自身的力量走到目的地

在我们辅导孩子学习中华学习机时，一定要注意发挥孩子的主观能动性。因为只有当他们积极思维、大胆实践和在经历失败与胜利的过程中，智力才得到提高。所以，靠孩子

自身的力量达到目的地——是最好的辅导方法。

我们和孩子一起操作中华学习机时，最好将自己也看作是一名初学者，和孩子一起商量、探索，并且要注意观察孩子有什么特殊的困难，以及他们擅长什么、喜欢什么，这样帮助他们就会有的放矢。在肯定他们正确的部分时，表扬不要过分，比如你说："你真是个了不起的孩子！""你真是个天才！"这样效果并不见得好，明智的孩子会感到言过其实，很不舒服；年龄小的孩子会盲目得意，撒娇。我们表扬要掌握分寸，主要表扬正确的做法而不是他本人。

对孩子错误的部分，要用商量的口气和实验的方法加以纠正。比如，你可以说："你看如果我们这样操作是不是会更好些？""你能这样操作吗？试试看！"有的爸爸妈妈"恨铁不成钢"，采用气极败坏的态度进行训斥："给你讲了多少次，还不会，真笨！"这样做会使孩子心中不满，或内疚，或愤怒，或仇恨顶撞，造成争执，大吵大闹，引起一系列不良反映。孩子和父母都不高兴，影响孩子的积极性。

另外，我们在辅导孩子学习中华学习机时，不要包办代替，该孩子操作的一定要他自己干，有时应该故意走开，让孩子独立操作，使他有个自由探索的机会，使学习更加主动'在孩子进行运算或画图时，不可避免地会出现错误或不妥，这时，学习机会马上出现反馈信息。例如，当运算答案不对、命令的格式不符合规定、画图设置的尺寸不合理时，学习机会告诉你："答案不对，是否要重作？"或者提示错误的类别，这时孩子会很乐意地检查自己的疏忽和错误，想办法纠正。这

比老师或家长去挑错．纠正效果要好得多。

传统的教育方法往往是过份追求尽少出现错误，并在实际出现错误时又忘记把它们当作学习的机会来对待。希望家长和老师在辅导孩子时要注意纠正这种做法，采用与学习机相适应的新的教育方法，充分利用学习机能迅速反馈的优点，给孩子以独立发现并纠正错误的机会。

总之，在学习的过程中，尽量调动孩子的内因，让孩子靠自身的力量走到目的地。

## 16.智力因素与非智力因素

非智力因素是指人的感情、意志、性格，如自信心、成就心理、力争意识、心理承受能力等。

事实告诉我们，有的人智力条件很好，常常由于非智力因素的影响，智力得不到发展，而不能取得什么成绩。

国外，有人对一千多名超常儿童追踪研究了30年之后，得出一条结论：智力和成绩之间绝不是完全相关的，20%成就最大的与20%成就最小的相比较，最明显的差别是这两部分人个性、意志、品质不同；前者自信，有进取心、不屈不挠，能坚持到最后完成任务，而后者却恰恰缺少这些品质。

有人对我国科技大学84、85届少年班的74名优秀少年大学生（平均15.5岁）和华东师大某系83届85名普通大学生（平均年龄19岁）进行的一次测试和比较发现：少年班大学生比普通大学生在个性、心理、品质方面，有明显优势：他们持强好胜，不甘落后，独立果断，比普通大学生更有耐力

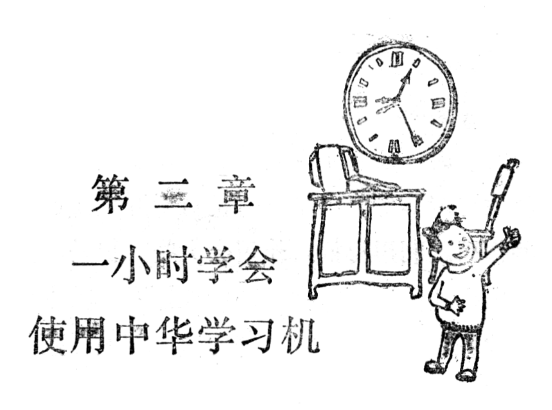
和韧性（看起书来可一连坐几个小时不受外界干扰），求知欲强。特别是少年班的优秀女大学生比普通女大学生怀疑性少，紧张性小，自信心强。所以说，发展智力不是孤立的，必须同培养良好的非智力因素相结合。

鉴于我国独生子女的现状，目前培养他们的非智力因素更是非常必要。一些孩子自信心差、胆怯害羞，有的意志薄弱、动摇不定，有的不能吃苦、害怕困难、学习兴致不高。所以，我们在辅导孩子学习时，要注重培养非智力因素，克服他们的心理障碍。

孩子的可塑性大，如果我们引导和教育方法得当，他们的心理障碍是不难克服的。比如，对于自信心差的孩子，我们要鼓励他在实际活动中取得成功，以此增强自信心。对于意志薄弱的，要鼓励他们克服困难，树立信心。人的意志力是在克服困难中锻炼出来的；中外有成就的人物都是在克服重重困难的逆境中成长起来的。我们不要为孩子将一切都准备得停停当当，要给他们留有克服困难的机会，使那些胆小、孤僻、优柔寡断的孩子受到锻炼，要能及时发现他们的微小进步，并给于肯定，帮助他们树立自信心和克服困难的勇气。

我们要为孩子创造一个有利于培养良好性格的环境，不仅是自然环境，主要是社会环境，让孩子在愉快、民主的气氛中不断提高非智力因素。中华学习机对此是可以起到积极作用的。老师和家长要抓住有利条件和有利时机，尽量做到自觉地培养孩子的非智力因素。

开发儿童的智力取决于多方面的因素，包括先天的影响、后天的教育，孩子个体的特殊性，社会、家庭、学校所起的作用，还有儿童实践的深度和广度，个人主观能动性发挥的程度等等。中华学习机对于开发儿童的智力能够起到一些作用。利用得好，作用会大些。但是，要有效地开发儿童智力应从多方面努力，不断地探索，其中也包括更广泛地利用中华学习机开发儿童智力这一新的课题。



# 第二章 一小时学会使用中华学习机

## 17.像学会使用收录机一样简单

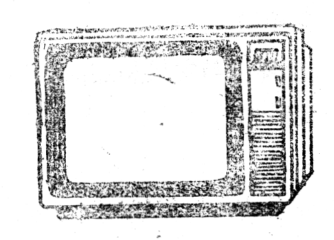
目前，不少人一提到计算机，就感到高深莫测，不敢问津。其实，学会使用中华学习机非常简单，就像学习使用收录机一样，中华学习机和其它先进设备一样，学会基本的操作，达到能够使用的目的是很容易的。

对于大多数人来讲，购置中华学习机不是为了培养软件程序员或者其他计算机专业人员，而是为了使用现成的软件来达到某种特定的目的。

本章正是为了满足这些使用者的要求，精选最基本的，也是必须掌握的操作方法，使工作繁忙的爸爸妈妈们只要花一个小时，就能与孩子一起享受电子游戏的乐趣；使功课繁重的学生花一个小时，就能请到一位耐心的“电子教师”，为他辅导功课。对于希望系统掌握中华学习机使用的同志，可参阅第五章。

## 18. 学会使用中华学习机玩电子游戏

目前，有许多游戏软件能够在中华学习机上运行，其中大多数都能存贮在录音机的磁带上。也就是说，录音机在这儿作为外存贮器。本节的目的就是要学会使用具有主机、家用PAL制式彩色或黑白电视机及录音机组成的中华学习机系统（图2-1），掌握基本操作方法。现在以游戏软件为例叙述。



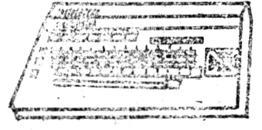




图 2-1

（1）将系统安装起来

打开中华学习机主机机箱，小心地取出主机，同时取出视频电缆和录音机电缆。

①主机：主机外壳右上角有一个电源指示灯，前半部是

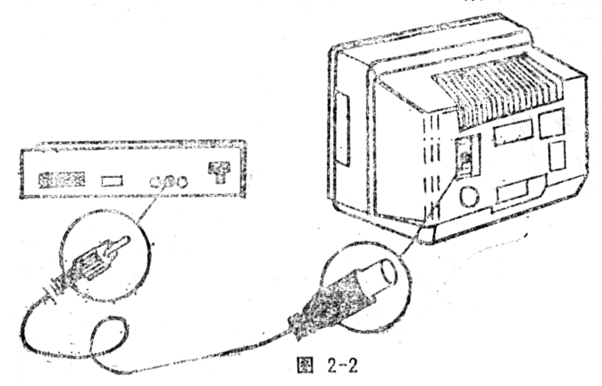


图 2-2

键盘，机壳内有主电路板、各种插件、电源、扬声器等。

②主机与显示器的连接：显示器可以是电视机(PAL-D制式彩色或黑白的），也可以是监视器(PAL-D制式单色或彩色的）。由于二者与主机之间连接的插口不同，要引起注意，以免接错。

A.电视机与主机相连：如图2-2把电视机的天线插头拔掉，用视频电缆线一头插入主机后座标有电视机（TV）的插孔，一头插入电视机天线插孔；将电视机电源插头与主机电源插头分别插入电源插座（220V交流市电）；先打开电视机电源开关，再打开主机电源开关；调整电视机频道（一般在 3或4频道），直到屏幕上清晰地显示出“ZHONG HUA XUE XI JI”字样为止。至此主机与电视机接线完毕。

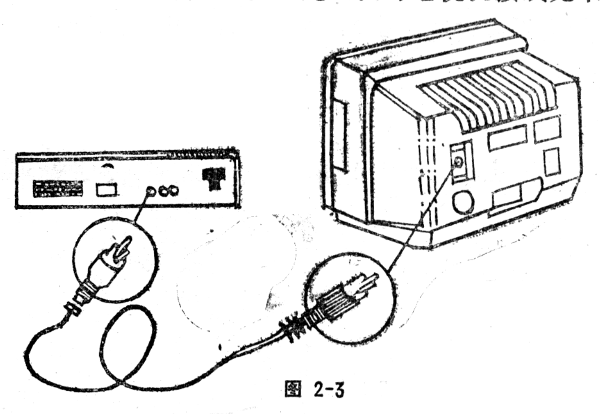


图 2-3

B.主机与监视器相连：如图 2-3，用视频电缆线一头插入主机座后监视器接口，一头插入监视器 IN 插座；将主机与监视器电源的插头分别插入电源插座；先开监视器电源开

关，再开主机电源开关，屏幕上将出现清晰“ZHONG HUA XUE XI JI”字样，至此，主机与监视器接线完毕。

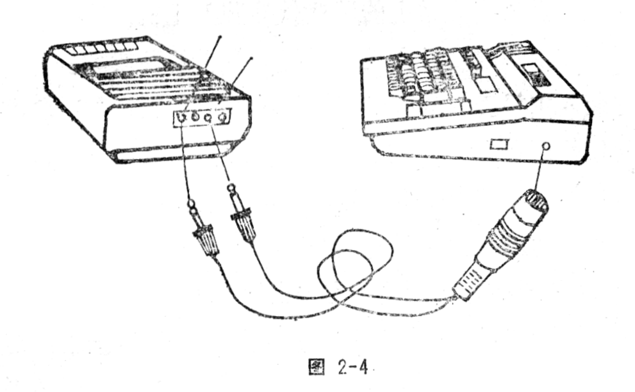


图 2-4

③主机与录音机连接：如图2-4，将录音机电缆线的一端五芯插头插入主机侧面的录音机接口，另一端的两个3.5mm插头分别插入录音机EAR插孔和MIC插孔。

这样系统便安装起来了。

(2) 熟悉一下键盘

为了能操纵中华学习机做游戏，需要大致熟悉一下键盘：找到26个英文字母、10个(0～9)数字键和符号键的大致位置（至于一些特殊的键，如果用到，可以查阅第五章的内容)。

(3) 开机启动

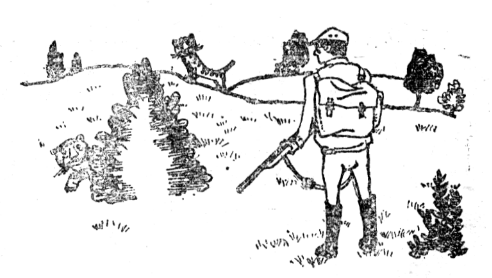
先打开电视机（或显示器）电源开关，再打开主机电源开关（在主机的左后方），立刻会听到“嘟”的一声，这说明主机已经准备好，同时键盘左上方的指示灯发亮，显示屏

上有一个闪烁的小方块，称为光标，“]”是BASIC语言系统提示符。

（4）将存贮在磁带上的游戏软件装入主机内存

要想在中华学习机主机上运行磁带上的游戏软件，必须先将它装入主机壳内的存贮器。请按以下步骤操作：将装有游戏软件的录音磁带放入录音机里（注意不要放反），把磁带倒回到原处，这时在主机键盘上敲入PLAY命令（方法是顺序按字母键P、L、A、Y）。在录音机上按下放音键(PLAY键)，使磁带开始转动；然后在主机键盘上找到|RETURN|键（它在键盘右部，叫回车键，以后用↵表示)，按下。这时屏幕上的光标消失，大约过15秒钟，出现“PLEASE WAITING—”（请稍等）之类的信息。当游戏软件全部装入内存后，屏幕上会出现五彩缤纷的游戏画面。最后按下录音机的STOP键（停止磁带转动）。

（5）开始玩游戏



现在就可以作游戏了。假如刚才我们调入主机内存的游戏软件名字为“打老虎”，这时，电视机屏幕上便会出现这样的画面：整个屏幕分为四层，层与层之间有几个梯子相连，三只小老虎到处跑来跑去，屏幕的下方站着一个小人。操作者要通过键盘使小人追上去，把老虎打死。键

盘上[A]键使小人向上跑，[Z]键使小人向下，[?]键使小人向左，[illegible]键使小人向右。通过按这四个键控制小人经过梯子登上楼层，然后选择老虎要经过的地方，用空格键（键盘最下方的长键）来指挥小人在楼板上挖坑，老虎一旦掉进坑里，再次按空格键，小人便会将老虎打死，屏幕右上方记分处便会加上100分。当把三只老虎全部打死，这一局取胜，中华学习机便会演奏一段优美的乐曲来庆贺胜利，并开始新的一局——老虎增加到五只；以后还会继续增加，并且老虎的运动速度也加快，而小人的速度却越来越慢，这对玩游戏者的要求就更高，兴趣会更浓。假若老虎撞上了小人，这一局便会以失败而告结束，要重新再来。

这只是游戏软件的一个例子，其它游戏的玩法会有所不同，可以从游戏磁带附有的说明中了解如何操作。

## 19.学会使用中华学习机辅助教学

课堂上辅助教学，课外帮助中、小学生复习功课（充当电子教师”)，是中华学习机主要的功能之一。目前这方面的软件很多，只要学会使用这些软件，便可以有助于我们的教学。

教学辅助软件多存贮在软磁盘上，当然，也有存贮在录音磁带上的，如果是后一种情况，便可以按照上一节“使用中华学习机玩游戏”的方法去进行。

软键盘的外形像一个小唱片，封装在保护套内，上面记录了大量的信息，其中包括教学辅助软件程序。要使用教学

辅助软件，就要将这些信息装进主机内存中；这些工作要通过软盘驱动器来完成。这样，就必须在原来的系统，即包括主机、显示器（电视机）和录音机的基础上，增加软盘驱动器。

这一节就是要通过学会使用存贮在磁盘上的教学辅助软件来了解具有主机、显示器（电视机）、软盘驱动器组成的中华学习机系统，掌握基本的操作要领（图2-5）。

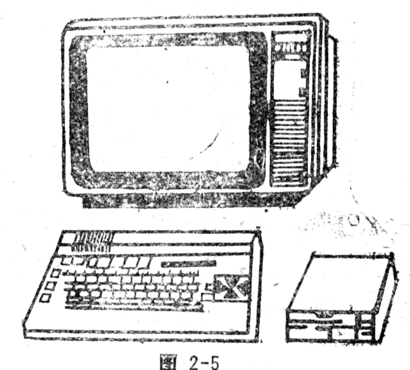


图 2-5

（1）把系统安装起来

上一节已经讲了如何将主机和显示器连接起来，现在只要讲讲如何把主机和软盘驱动器连接起来就行了。

软盘驱动器本身与唱机相似，由一个马达带动磁盘转动，机上有一个磁头可以读取磁盘上的信息。

软盘驱动器和主机之间的连线方法如图 2-6。它们之间是通过20线扁平电缆线相连的。电缆的一端插入主机后驱动器接口，另一端连接驱动器；连接时一定要注意，插针和插孔必须一一对应，即电缆线两端的凸起部分都朝上。

（2）开机启动与软件的装入

系统安装好后，就要开机启动了，同时将磁盘上的教学辅助软件调入主机内存并且运行。

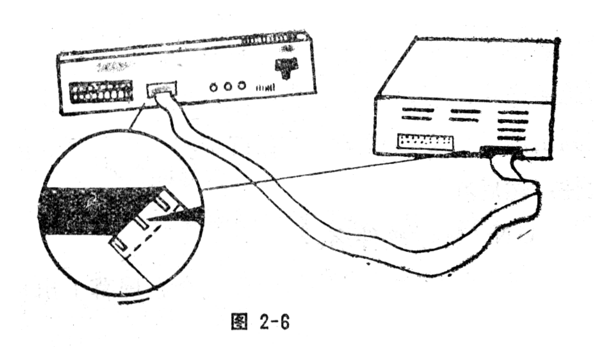


图 2-6

首先，打开软盘驱动器正面的一个小盖，将教学辅助软件磁盘印字面向上、有缺口地方向里轻轻地插入驱动器，关上驱动器门盖，然后依次打开显示器电源开关、主机电源开关，这时会看到驱动器上指示灯发亮，并且可以听到驱动器内电机转动的声音，当驱动器指示灯熄灭后，如果屏幕上出现教学辅助软件的画面，如标题录信息等，说明教学辅助软件已经从磁盘调入主机内存了并且开始运行；假若当驱动器指示灯熄灭后，屏幕上出现如下信息：

```
DOS VERSION 3.3                    08125180
APPLE II PLUS OR ROMCARD  SYSTEM
MASTER
(LOADING INTEGER ANTO LANGUAGE
CARD)
]□
```

这时，我们可以从键盘上键入 CATALOG ↵命令（即

顺次键入字母键 C、A、T A、L、O、G 和 RETURN 键）。这是一条"列磁盘文件目录"的命令。当学习机接受到这条命令之后，屏幕上会出现此磁盘上所有文件的目录，在列出的这些文件目录中，你可选择出准备运行的教学辅助软件，然后在键盘上键入：

RUN 教学软件名↵

此时，教学辅助软件的画面便会在屏幕上出现了，这说明此教学辅助软件已经从磁盘调入内存，并且开始运行。

（3）使用教学辅助软件

目前，使用中的教学辅助软件大多都具有直观、明了、使用方便等优点，这是由于在设计这类软件时，对每一操作步骤都安排了提示信息。例如，在程序运行开始，屏幕上一般都会出现一个功能表（称之为菜单），供使用者根据需要实现的功能键入相应的编号；这样，即使没有任何计算机知识的人，只要跟着提示信息做出相应的回答，便可以一步步地进行下去。

例如，解析几何作图教学辅助软件是针对高中数学中有关解析几何作图的几个难点而设计制作的，可以帮助教师在讲课时画出复杂的轨迹方程图，同时，也帮助学生形象地理解由于参数变化所引起的轨迹的变化。

这个软件按上述第一种情况开机并调入内存后屏幕上出现主菜单：

①任意方程的轨迹图

②动点关于点的轨迹图

③动点关于已知线的轨迹图

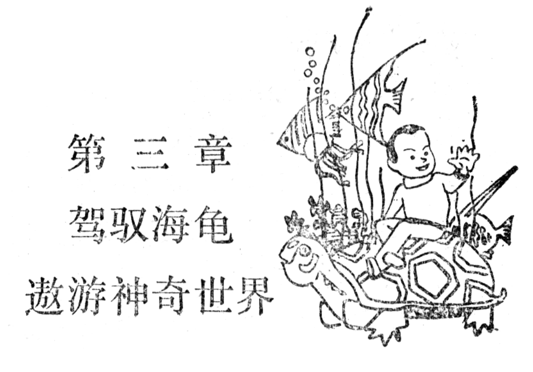
④动点关于已知点和已知线的迹轨图

⑤动点关于已知线和已知线的轨迹图

⑥点关于三角函数的轨迹图

使用者根据需要选择其中的一种功能，从键盘上键入其编号，如键入③↩，计算机便会自动进入这种轨迹的作图状态，然后会继续提示让你输入数据、符号等。你只要按提示操作，便可以一步步地实现预定的各种功能，包括将画出的图形扩大或缩小等等。

教学辅助软件很多，我们不能一一列举，只要掌握基本操作方法，再参照使用说明，便可使用中华学习机去完成教学辅助工作了



# 第三章
# 驾驭海龟
# 遨游神奇世界

在海边沙滩上爬行的海龟，会在沙滩上留下各式各样的痕迹。而我们这里的海龟却是计算机显示屏幕上的一个亮三角，我们利用LOGO语言提供的各种命令，能使它在屏幕上留下各种各样的图形。

LOGO语言是计算机高级语言家庭中的一个新成员。

LOGO一词起源于希腊单词，意思是符号，或思考，或想法等。它作为一种高级语言起源于麻省理工学院著名的人工智能研究中心，是目前计算机发达国家的中小学普遍采用的语言。它打破了过去传统的教学方式，采用启发式学习方法循循善诱地引导学生们去思考。不但小学生，就是五六岁的儿童都可以在电脑上作饶有风趣的游戏——凭着自己的想像去绘制各种图画。这真难以想象它的潜力该有多大。因而，LOGO语言被誉为“电脑教室里的国王”。

另外，LOGO语言是先进的结构程序设计语言。这样，不仅能使孩子们从小就学会良好的程序设计技术，为以后的

学习打好基础，而且还可以培养孩子结构化思维方法。

海龟作图是LOGO语言的一部分。这一部分内容简单易学，对低幼儿童非常适宜。儿童不仅可以学会操纵海龟在屏幕上绘制和“装配”各种图形，而且还能使图形具有各种变幻，甚至使它动起来。

因此，在学习这部分内容时，我们建议家长和孩子一起学习，放开手，让孩子尽兴地“玩”海龟。而家长的任务是事先掌握这部分内容，然后坐在孩子旁边，观察他的操作，提一些建议。有时候要故意让他独立操作一会，而有时又要和孩子共同探讨、解决问题。

总之，要使孩子能保持极大的兴趣，展开想像的翅膀，大胆地创作。这样才能在学习程序设计的过程中达到开发智力的目的。

## 一、海 龟 世 界

我们的主角——海龟就要与大家见面了。本章的目的就是通过学习一些命令使海龟移动，并在电视屏幕上绘画作图。

### 20.海龟出场

中华学习机已经将LOGO语言固化在主机电路板上的只读存贮器内。它基本具有 Terrapin LOGO的功能。对于不同的机器配置，可以采用不同的方法将 LOGO 语言调出。

当你的中华学习机系统只配有主机、电视机和录音机的

情况下，先将电视机和录音机电源打开，再打开主机电源开关，当屏幕左下方显示出提示符“]”时，键入

]LG↵

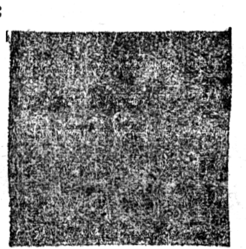
即可进入LOGO语言。

当你的中华学习机配有软盘驱动器的情况下，首先将标有DOS3.3的磁盘插入软盘驱动器。注意在插入弹盘时要将有标签一面朝上，缺口朝里。关上驱动器门柄，然后依次打开电视机和主机开关，此时听到驱动器转动的声音，几分钟后，屏幕上出现提示符“]”时，键入：

]MAXFILES1

]LOGO

便会出现以下版面：



CHINESE EDUCATION COMPUTER

LG

VERSION1.1

1987.6

?

其中“？”是LOGO语言的提示符，至此已将LOGO语言启动

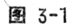
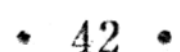


图 3-1

LOGO语言可以指挥一个叫做海龟的自动装置，通过命令，让它在屏幕上移动，画出各种图形。

好，现在我们要请海龟出场了！

键入：



DRAW↲

此时整个屏幕变得一片空白，只有中央有一个小三角，这就是所谓的海龟，靠近底部有一个问号提示符。这意味着LOGO进入作图模式。这时屏幕底部保留4行以供你键入的命令和计算机作回答时用。屏幕的其余部分均供作图(图3-1)。

下面我们将通过不同的命令，使海龟移动画出各种图形。

## 21.让海龟画出正方形和三角形

——介绍海龟基本命令

上一节海龟已经出场，它的原始位置是位于屏幕的中央，并且尖头向上（图3-1）。

尖头方向表示海龟的前进方向。

以下是使海龟移动的基本命令：

| 向前：FORWARD | （缩写FD） | 参数 |
|---|---|---|
| 向后：BACK | （缩写BK） | 参数 |

这两个命令是让海龟向前或者向后移动，附带的参数是确定海龟前进或者后退的步数。

整个屏幕左右为280步，上下为200步左右；若从屏幕的中心点(0，0)算起，到屏幕最左边为140步，到屏幕最右边为139步，到最上边为101步左右，到最下边为100步左右，例如：

FD 40 ↲ 即命令海龟向前40步（当然在键入命令以后，要键入回车键，这我们以后就不加说明了）。

以下命令是使海龟左右转动的命令：

| 向左：LEFT | （缩写LT） | 参数 |
| --- | --- | --- |
| 向右：RIGHT | （缩写RT） | 参数 |

它们附带的参数是确定海龟转动的角度。例如：LT 90 即命令海龟向左转90度。这时海龟并没有走动，只是从原来前进的方向向左转了90度，如果原来前进方向为向上，执行此语句后，海龟方向指向正左方（这里与数学上的规定相同，转一周为360度）。

现在我们就可以画正方形了。观察每键入一个命令后，海龟的移动方向和距离。

FD 50
RT 90
FD 50
RT 90
FD 50
RT 90
FD 50
RT 90

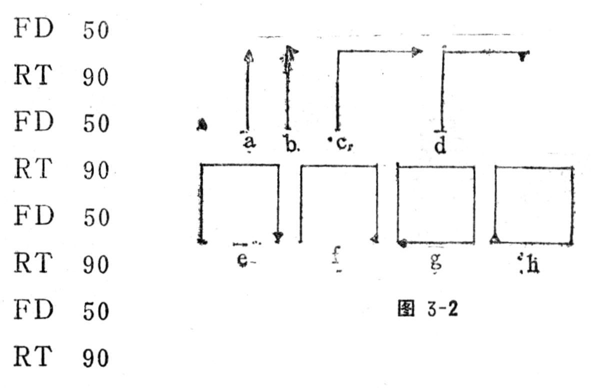


图 3-2

图 3-2 是以图解的方式来描述执行过程：a.前进50步；b.右转90度；c.前进50步；d.右转90度；e.前进50步；f.右转90度；g.前进50步；h.右转90度（回到起始位置）。

我们不难用同样的方法画出等边三角形：

FD 60
LT 120

FD　60

LT　120

FD　60

LT　120



图 3-3

确定转动的角度是海龟作图的一个关键问题，在这里转动的角度是相对于原来的方向而言的。

从上面两个图形的作图过程中，可以看到我们所键入的命令重复很多，画正方形时，前进和右转命令重复了四次；画三角形时，前进和左转的命令重复了三次；这样做有些太烦琐了。用“重复”命令可以简化这些操作。

以下是重复命令格式：

| REPEAT　参数〔命令表〕 |
|---|

其中，参数为重复次数；命令表为重复的内容（要用方括号括住）。

〔例〕　REPEAT　4 [FD 50 RT 90]

即将FD　50　RT　90　重复执行4次；于是只用一条语句便将正方形画了出来。

同样画三角形的命令也可以用一条命令来代替：

REPEAT　3　[FD　60　LT　120]

请你试一试语句：

REPEAT　5　[FD　50　LT　72]

会作出什么图形。

根据上面语句，海龟画出一个如图 3-4 的正五边形。从三角形、正方形、正五边形的画法上，我们是否看到了一个规律，海龟从开始指向上方，每次转动一定角度，最后又回到开始的方向，共转动 360 度；我们用 360度除以转动的次数就得到了转动的角度；转动的次数（即重复次数）是所画多边形的边数。

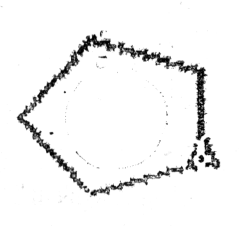


图 3-4

〔例〕 REPEAT 10 [ FD 30 RT 360/10]

键入此命令后，屏幕上的海龟便画出一个正十边形。

## 22.让海龟按照座标来画图

刚才，我们用前、后、左、右四个命令画出了各种图形，而且又用重复命令加以简化。但是，如果我们想在屏幕上某一位置画一图形，或者将海龟移至屏幕上某一处，就要计算出行走的距离和转动的角度，这样就显得格外麻烦。如果引入座标，问题就会变得简单多了。

(1) 设置海龟的位置

对于没有学过座标的人来说，要知道屏幕上的一个点就如同电影院里的某一个座位一样，是由排、座号来确定的，即屏幕上的每一点是用一组顺序的数来表示的。屏幕上的座标这样规定：通过屏幕中心画一条从左到右的水平线，这条水平线以步为单位标上刻度，中心点为 0，向右为正值（最大为 139 步），向左为负值（最小为 --140）。这条带刻度的有

方向的水平线叫X轴。同样，通过中心点画一条与X轴垂直的线，这条线也以步为单位，划上刻度。从中心点向上定为正值；从1步到100步。从中心点向下为负值，从－1到－100步，这条直线叫Y轴（图3-5）。

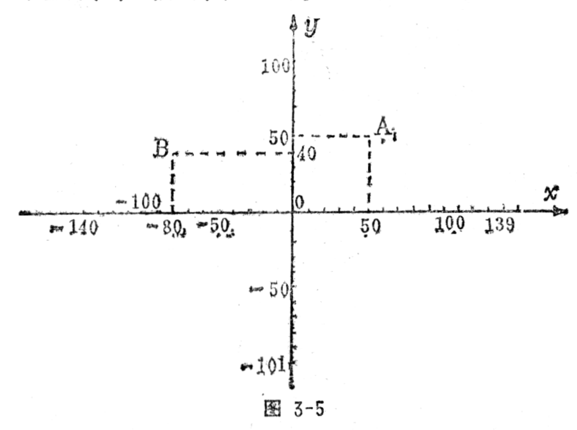



图 3-5

这样屏幕上的任何一点都可以用座标来表示了。如图3-5中A点，它对应X轴座标为50，Y轴座标也为50，于是便可表示为A(50，50)；B点则要表示为B(－80，40)。括号里两个数值中前一个表示X轴的座标值，后一个表示Y轴的座标值，不可搞颠倒了。

那么我们如何将海龟移动到A点或者B点呢，这要用到以下两个命令：

| | |
|---|---|
| SETX | 参数 |
| SETY | 参数 |
| SETXY | 参数1 参数2 |

其中：参数为座标值；SETX是将海龟移至参数值所指

的X；SETY是将海龟移至参数值所指的Y。SETXY 是 为将海龟移至参数1和参数2所指的（X，Y）座标位 置上。

例如：现在要将海龟移至A点或B点，就用不着计算要走多少步，转动多大角度。直接用上面给出的命令即可。

将海龟移至A点的命令：

SETX 50

SETY 50

或SETXY 50 50

将海龟移至B点的命令：

SETX －80

SETY 40

或SETXY －80 40

这样无论海龟原来在什么位置，执行移位命令后，便会移至指定地点了。

这两条命令是确定海龟的绝对位置，并没有改变海龟的方向。即假如原来海龟在初始位置方向是向上的，移至A点后方向依然向上。

（2）设置海龟的方向

有时希望将海龟设定指向某一角度而与原来的指向无关，这时要用一个规定：0或360度为海龟指向正上方；180度指向正下方；90度指向正右方。则用如下命令来设置海龟的方向：

| SETHEADING 角度（缩写SETH） |
| --- |

如 SETH 270 则表示海龟指向正左方。

(3) 如何知道海龟的位置与方向

当我们画一个复杂图形时，由于海龟转来转去，使我们一时不清楚海龟的位置和方向。而命令：HEADING、XCOR和YCOR，可以让计算机告诉我们海龟目前的位置和方向。

当我们键入

```
PRINT HEADING XCOR YCOR
```

屏幕上立即显示出：180 −80 40

意思是海龟目前的方向是指向正下方，位置为（−80，40），即X座标值为−80，Y 座标值为40，即图3-5的B点。

**23.美丽的色彩**

如果你的显示器是彩色电视机或者彩色显示器，你便可以在不同颜色的背景下，画出彩色线条。

命令如下：

```
PENCOLOR(缩写PC)      颜色编号  （彩笔）
BACKGROUND(缩写BG)    颜色编号  （背景）
```

其中颜色编号如下：0——黑色；1——白色；2——绿色；3——紫色；4——桔黄色；5——蓝色。

如果你没有明确定义背景和彩笔的颜色，LOGO将自动地采用背景为0（即黑色），而彩笔则自动地采用1（即白色）。

另外，在选择色彩时也有一些规定：在桔黄色或者蓝色背景上不能用绿色或紫色的彩笔。在绿色和紫色的背景上也不能用桔黄色或蓝色的彩笔。

如果你在屏幕上作画，然后改变背景颜色，那么画中各行的色彩也会改变，或者会产生奇妙的效果。有些人喜欢用这些效果来实践。而最好的办法是，如果要改变背景颜色，先要清除屏幕。清除屏幕的命令：CS↲。

为了获得清晰的色彩，LOGO系统画出的线条都比较粗，如果你不考虑颜色效果，采用背景6和彩笔，则能使海龟在显示屏幕上画出漂亮的细线。

表3-1列出在不同号背景上作画时，不同号彩笔的效

表 3-1

| 彩笔号 \ 效果 \ 背景号 | 0 | 1 | 2 | 3 | 4 | 5 | 6 |
|---|---|---|---|---|---|---|---|
| 0 | 黑色 | 黑色 | 黑色 | 黑色 | 黑色 | 黑色 | 黑色 |
| 1 | 白色 | 白色 | 白色 | 白色 | 白色 | 白色 | 白色 |
| 2 | 绿色 | 绿色 | 绿色 | 绿色 | 桔黄 | 桔黄 | 白色 |
| 3 | 紫色 | 紫色 | 紫色 | 紫色 | 蓝色 | 蓝色 | 白色 |
| 4 | 桔黄 | 桔黄 | 绿色 | 绿色 | 桔黄 | 桔黄 | 白色 |
| 5 | 蓝色 | 蓝色 | 紫色 | 紫色 | 蓝色 | 蓝色 | 白色 |

果。

下面我们用不同的颜色画一个正方形，而它的四条边分别为黑色、绿色、紫色和桔黄色，而背景用白色：

```
BG  1
PC  0
FD  50
RT  90
PC  2
FD  50
RT  90
PC  3
FD  50
RT  90
PC  4
FD  50
RT  90
```

## 24.让海龟学些新本事

前面我们已经介绍了海龟作图的基本命令，并画了一些简单图形，现在假如我们想在正方形中间再画一个小正方形，而且两个正方形之间不留痕迹的话怎么办呢？

这里提供一个抬笔的命令和需要画线时的一个落笔的命令：

PENUP　　（缩写PU）　抬笔

PENDOWN　（缩写PD）　落笔

这样，当我们将大正方形画好后，下达抬笔的命令使海龟走到小正方形的位置，然后再下达落笔的命令，画出小正方形，两个正方形之间就不留任何痕迹了。

具体如何画，大家自己在电脑上试一试。

当我们在画较为复杂的图形时，海龟转来转去，有时海龟挡住了我们的视线，看不清究竟要转多大的角度，这时我们可以将海龟暂时隐藏起来，即键入藏龟的命令；需要时再将海龟显示出来。下达的命令为：

HIDETURTLE　（缩写 HT）藏龟

SHOWTURTLE（缩写 ST）显龟

在我们画完一个图形，在画另一个图形之前，需要将屏幕清干净，并且将海龟移回到原点——屏幕的正中央，此时键入清屏命令：

CLEARSCREEN（缩写CS）清屏

当我们键入 CS ↵之后，屏幕就被清除干净，海龟移至中央。

## 二、进一步的海龟课题

### 25.让学习机记住海龟的作品

前几节我们在用海龟作图时，是通过不断地下达命令让海龟完成的。每键入一条命令，按完回车键后，计算机就会立即执行，这种执行方式称为“立即执行方式”。这种方式计算机等待时间长，而且当执行完这条命令后，图形就会从计算机上消失了。例如画正方形，我们将许多条命令一一键入，屏幕上会画出现一个正方形。而假如说我们要画几个正方形，或者在另一个图上需要正方形时，又要重新将这一串命令键入，比较烦琐。而LOGO语言却能教会学习机画这些图形，并且存起来，当我们在需要的时候可直接从“记忆”中取出。

下面就让我们先教学习机来画正方形：

首先给要画的这张图起一个名字，但不能与LOGO命令的名字相同。这里我们就选正方形三个字的汉语拼音字头ZFX吧。

要教会学习机画正方形，必须键入：

```
TO  ZFX
FD 50
RT 90
FD 50
RT 90
```

```
FD 50
RT 90
FD 50
RT 90
END
```

第一行 TO ZFX 就是告诉学习机，要教给它一个新命令就是 ZFX，接下来八行是怎样画正方形，最后一行END是告诉学习机教它画正方形的过程结束了。

最后键入CTRL-C。屏幕上出现

ZFX DEFINED

这时，我们键入 ZFX↙，屏幕上会立即画出一正方形，说明学习机已经学会并记住了如何画正方形了。

像上面那样由若干条LOGO命令组合在一起，完成某一特定任务的经过，我们称之为过程。在使用上，就像一条命令一样。

过程的整体形式是：

TO 〈过程的名子〉 ……过程头；

过程内容 ……过程体；

END ……过程尾。

当键入 TO 过程名后，提示符“？”消失。说明现在处于一个新的模式，称为程序定义模式。当处于程序定义模式时，键入的命令计算机并不执行，而是将正在定义的过程贮存起来，当最后键入END和CTRL-C后，计算机以 LOGO 语言给出一个信息作为回答：

ZFX DEFINED

表明过程已经定义完毕，并退出定义模式。这时当我们键入ZFX↲就能画出一个正方形来。

同样的方法我们可以教会学习机画三角形：

```
TO SJX
REPEAT 3 [FD 60 LT 120]
END
```

这时学习机便学会画三角形了。于是，我们便将SJX作为新的LOGO命令用于进一步的定义中。

## 26.编辑

假如一个过程已被定义和使用，但又感到不怎么满意，这时可以对它进行修改和编辑。当我们键入EDIT和过程标题之后，系统便进入编辑模式，此过程的内容会在屏幕上列出，这时就可以修改。

我们以过程 ZFX 为例。若键入：

```
EDIT  ZFX
```

计算机进入编辑模式，ZFX过程被列出：

```
TO ZFX
FD 50
RT 90
FD 50
RT 90
FD 50
```

```
RT 90
FD 50
RT 90
END
```

这时光标在屏幕的左上角T字段上。下面我们介绍经常用到的几个编辑命令供过程的修改和编辑用。

（1）移动光标

下面这些键能将光标上、下、左、右移动，但不改变任何字符：

CTRL—P　光标向上移动一行；

CTRL—N　光标向下移动一行；

◁　光标向左移动一格；

▷　光标向右移动一格。

一般我们使用上述键是为了将光标移至要修改的字符处等待修改。我们可以试一试，比如我们想将上例中所有的RT改为LT，于是我们连续用CTRL—N，将光标移至第一个“R”处。

请记住：键入上述键时，一定要按住CTRL键不放松，然后再按N键或P键。

（2）删除光标处字符

CTRL—D键，可以擦除光标所在位置的字符，如：

RT 90




按CTRL—D会出现：

T 90
↑
光标

再键入L，有

LT 90。
↑
光标

以上操作实际上是将R改为T。

（3）在行的中间插入字符

需要在行的中间插入一个或多个字符时，需将光标移到待插入字符的位置，这时可键入需要插入的字符。如：

LT 90
↑
光标

键入 EF 后将得到：

LEFT
↑
光标

光标前的EF就是刚插入的字符。将光标移到E上，按两次CTRL—D键，即可删去EF。

（4）在已有的行间插入新行

移动光标至准备插入的一行的上一行行首，按下CTRL—O键，会在此行下面出现一空行。

LT 90
↑
光标

按 CTRL—O 则打开空行供插入用：

LT 90

光标

FD 50

现键入：

AA↵

将变为

LT 90

AA

FD 50

这样，在两行之间便插入一新行，内容为AA。

（5）删除光标左侧的字符

如果想删除光标左侧的字符，可利用 ESC 键起到橡皮的作用。这样，可以将我们键入的某个错字符擦去。

（6）退出编辑

退出编辑状态一般分为这两种情况：

①不保存编辑结果，退出编辑：有时，当你进入编辑模式后会突然改变主意，此时键入CTRL—G 后，便会退出编辑状态，刚键入的任何编辑命令都会作废，即所作的一切修改无效。

②保存编辑结果，退出编辑状态：当我们经过编辑修改，认为满意后，键入 CTRL—C，便会将编辑结果保存起来，并退出编辑状态。

编辑命令小结

CTRL—P　光标上移一行；

CTRL—N　光标下移一行；

◁　光标左移一格；

▷　光标右移一格；

CTRL—D　删除光标处字符；

CTRL—O　增加一空行供插入；

CTRL—G　不作贮存退出编辑；

CTRL—C　保存结果退出编辑。

## 27.新的探索

前面我们已经教会电脑作一些图形——定义过程，但能不能在不改变过程内容的前提下改变图形的大小和形状呢？回答是肯定的。

(1) 改变图形大小的输入参数

看图 3-6。这个图案是四个大小不同的正方形，现在我们要用同一个过程来实现它。

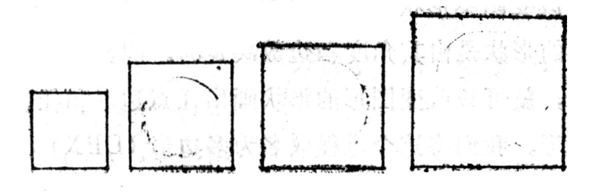


图 3-6

```
TO ZFX :x
FD :x
LT 90
```

```
FD :x
LT 90
FD :x
LT 90
FD :x
LT 90
END
```

或者也可以用这种方法来实现

```
TO ZFX :x
REPEAT 4 [FD :x  LT90]
END
```

其中x 称为变量，在这个例子中它代表了正方形的边长。要想画出不同边长的正方形，就要在执行时送入不同的数值。例如，我们先画出上图中最小的正方形，即可键入ZFX 30；而ZFX 60 就画出一个大正方形来。

（2）改变图案形状的输入参数

图案的形状是由其角度和边数决定的，因此改变这两个输入参数，就可以改变图形的形状画出任意边数和任意长度的正多边形。我们将这个过程取名为多边形（DBX），用 x 作为边长变量，用 a 作为角度变量：

```
TO DBX :x  :a
REPEAT 360/:a [FD :x  RT :a]
END
```

其中：360/:a 是计算边数，因为正多边形每个角相等，

而旋转一周回到起始点为360度，当 a＝90，360/90＝4，画出四边形；当 a＝72，360/72＝5，画出正五边形(图3-7)。

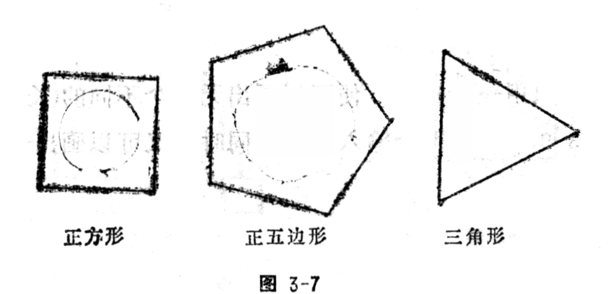
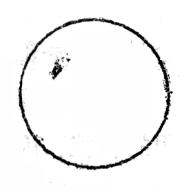


图 3-7

用这个过程还可以画出一个圆，即键入 DBX 1 1（图3-8）。


图 3-8

这个圆可以看作由许多非常短的边组成的。前进一步，旋转一度，便可以画出一个圆来。

需要有两个输入参数的另一个图形例子是长方形，它需要有两个不同的尺寸与长和宽相对应。

取过程名为 CFX（长方形），用 a 变量表示长，用 b 变量表示宽。过程如下：

```
TO CFX :a :b
FD :a
RT 90
FD :b
RT 90
FD :a
```

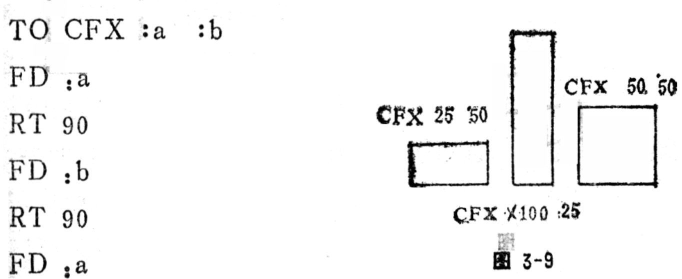


图 3-9

```
RT 90
FD :b
RT 90
END
```

仅仅用一个过程，便可以画出无数个不同的长方形来（图 3-9 ）。当两个输入参数相同时，又可以画出许多正方形。

## 28.过程和子过程

教会学习机的一个命令叫做一个过程，而一个过程就象LOGO的原始命令一样可以被使用，当一个过程用来作为另一个过程的一部分时，就把它叫做子过程。

使用子过程来画图，可以说是LOGO语言中最重要、最有用的功能。因为许多极其复杂的图形，都是由一些基本图形组成的。若用子过程来表达一些基本图形，会使许多看起来非常难画的图变得异常简单。如图3-10～13等有趣的图

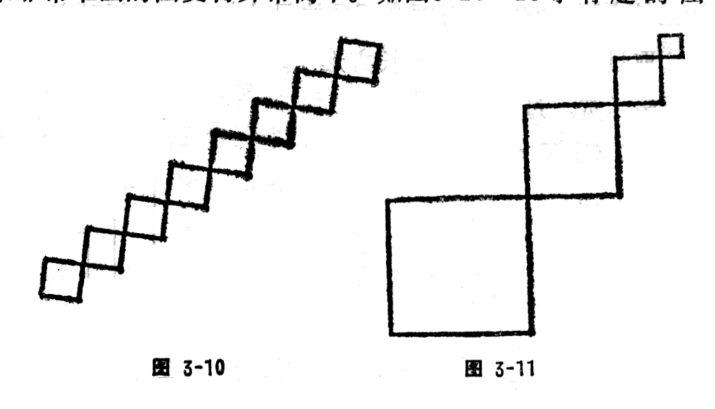


图 3-10　　图 3-11

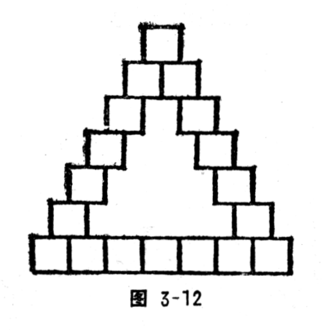


图 3-12

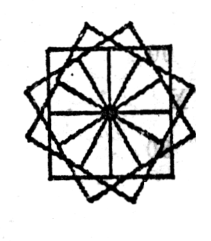


图 3-13

形，都是由正方形组成的。

我们还可以将画正方形的过程(ZFX) 作为 LOGO 语言中的一个命令来使用，就如同 FD、RT一样。但ZFX的作用是画出一个正方形。若加上参数还可以画出不同边长的正方形。

现在，我们准备将 ZFX 作为一个子过程，来画一扇窗户（图3-14）。

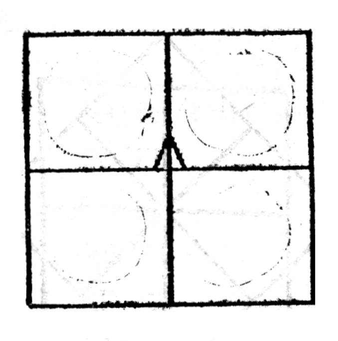


图 3-14

这扇窗户是由四个大小相同的正方形组成的。正方形过程已经教会学习机了；如果还没有教会它，只要键入本章“25.”中ZFX的过程就可以了。

在学习机已经学会了如何画正方形以后，可以开始画窗户了。我们起名为(CH)：

TO CH

```
ZFX
LT 90
ZFX
LT 90
ZFX
LT 90
ZFX
LT 90
END
```

即画一个正方形就让海龟旋转90度，直到海龟又回到起始位置与起始方向，一扇窗户的图案便完成了。

当然，你如果想继续画下去，的确是一个好主意；可将窗户作为一个子过程，再组成新的图案，如图3-15。想一想，这个过程该如何写呢？

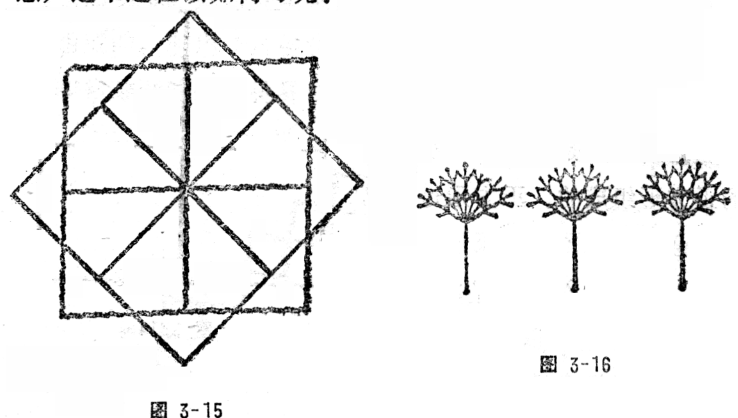


图 3-15

图 3-16

下一个运用子过程的典型例子是画一片小树林（图3-

16）。

小树林是由一棵棵小树组成的，小树由丛、枝、叶组成（图3-17）。

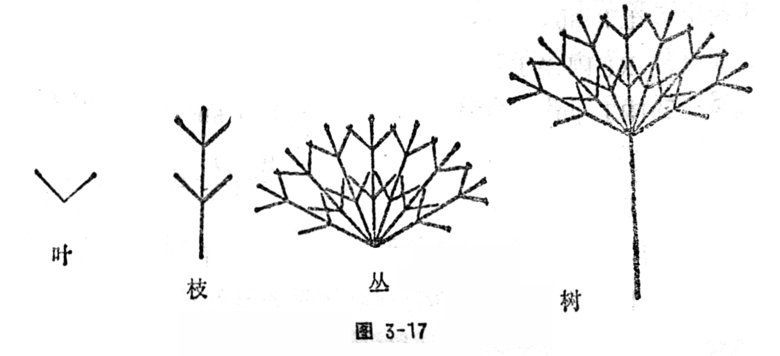



图 3-17

先将“叶”的过程编写出，编写“枝”过程时可以调用“叶”过程，编“丛”过程时，可以调用“枝”过程，编“树”过程时可以调用“丛”过程，编“林”过程时便可以调用“树”过程了。

(1) 叶过程（YE）

```
TO YE
LT 45
FD 10
BACK 10
RT 90
FD 10
BACK 10
LT 45
```

(2) 枝过程（ZHI）

```
TO ZHI
FD 15
YE
FD 15
YE
FD 10
BACK 40
END
```

```
END
```

(3) 从过程 (CONG)

```
TO CONG
LT 60
REPEAT 7 [ZHI
RT 20]
LT 80
END
```

(4) 树过程 (SHU)

```
TO SHU
CONG
BACK 60
FD 60
END
```

最后我们就可以很快地写出“林”的过程：

```
TO LIN
PU
LT 90
FD 70
RT 90
PD
REPEAT 2 [SHU PU RT 90 FD 70 LT 90 PD]
SHU
END
```

也可以将画树与树之间的间隙作为子程序写一段过程(YI)：

```
TO YI
PU
LT 90
FD 70
```

```
TO YIN
REPEAT 3 [SHU YI]
```

```
RT 90
PD
END
```

这样的程序设计多么像堆积木呀！一个大的模型由若干个简单的积木块构成。这就是LOGO语言的特点，这种设计方法就称为结构程序设计，它为程序设计开创了新的一页。

## 29.海龟的作品

通过前面的学习，我们可以看到海龟具有各种各样的本领，现在，该是让海龟给我们表演的时候了。

（1）画一座房子（如图3-18）

先来分析一下房子的结构：它是由房顶、墙壁和门组成。下面我们就像盖房子一样，将它们分别做好，然后再安装起来。

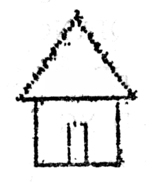


图 3-18

先画墙壁，它实际上是一个正方形，我们可以将已定义好的正方形（过程ZFX）直接使用：

```
TO QIANG
ZFX 25
END
```

门实际上是一个小的长方形，那么可将画长方形的过程CFX直接使用：

```
TO MEN
```

```
CFX 15 5
END
```

房顶实际上是一个正三角形，则三角形过程SJX可直接使用：

```
TO DING
SJX 35
END
```

各种部件均已做完。下一步就是要如何安装了。安装可要有一定的技术，否则会装颠倒，或者接不上，房子显得不能用或不安全。所以，一定要考虑周密。先将墙竖起来，走到门的位置，装起门，然后再盖房顶。

```
TO FANG
QIANG RT 90  FD 10  LT 90  MEN  LT 90
FD 10  RT 90  FD 25  LT 90  FD 5  RT 180
DING  END
```

至此，我们的小房子便盖起来了。

从画小房子的过程，我们可以看出海龟画图和我们平时画图有很大的区别；学习机必竟是机器人，它的一举一动全靠人来指挥。即使有点错误，它也不会帮你纠正。

（2）画一个小人

图 3-19

前面海龟已经为我们画了一座房子，现在海龟还要再为我们画一位此房子的主人，就像图 3-19

一样，有头、身子、手臂和腿。

①身子最简单，先来完成它：

```
TO SHEN
FD 50
END
```

②头：我们按照先画脸庞，再画嘴、鼻子、最后画眼睛。脸庞是一个圆，其余由线段组成。

```
TO TOU
LT 90 REPEAT 72 [FD1 RT5] RT 90 PU
FD 3 LT 90 PD FD 3 BK 6 FD 3 RT 90
PU FD 2 PD FD 5 PU FD 2 LT 90 FD 3 PD
FD 1 PU BK 8 PD FD 1 PU FD 3 LT 90
FD 12 PD
END
```

③手臂：

```
TO SHOU
LT 90 FD 30 BK 60 FD 30 RT 90
END
```

④腿：

```
TO TUI
LT 135 FD 40 BK 40 LT 90 FD 40 BK 40
LT 135
END
```

⑤将头、身子、手臂和腿安装起来：

```
TO REN
SHEN TOU FD 5 SHOU FD 45
LT 180 TUI
END
```

当我们再次键入 REN↵后，一个小人便会出现在屏幕上了。

## 三、更深入的海龟课题

这一章我们将进行更深入的海龟课题，即通过重复、递归等手段，使海龟画出更加奇妙的图形来。

### 30.美丽的螺旋线

在上一章里我们学习了过程和子过程。所谓子过程，就是一个过程被另一个过程所调用。那么假如一个过程调用其自己本身，即把本身作为子过程，那将出现什么情况呢？

让我们来做一次试验吧，我们把这个过程起名为LUO：

```
TO LUO :x
FD :x LT 90
LUO :x+2
END
```

当我们键入：LUO 5 ↵

在我们面前便会出现一个漂亮的螺旋线（图3-20）。

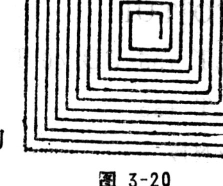


图 3-20

那么它是如何形成的呢？下面让我们分析一下 海龟的执行过程：

海龟首先执行 FD :x LT 90 前进5步，左转90度，这时，下一语句为 LUO :x+2，这就意味着要返回来调用自己本身，但参数增加了2，即 x=7。于是下一步就要前进7步，左转90度，又执行到 LUO :x+2，于是又在7步的基础上增加2，前进9步，左转90度，……。就这样反反复复地一直执行下去。

你也许会问："海龟能自己停下来吗？"这个问题问得好。但回答是："它自己不会停下来！"

如果要想使海龟停下来，可以按 CTRL—G 键。

能不能寻找一种方法预先控制海龟在合适的地方停下来呢？可以。如果设置一个关卡，当海龟通不过关卡时，就用 STOP 命令，使它停下来。

在上面例子中，每递归调用一次，边长x 增加2，如果命令当 x>50时，海龟应停下来，那么，这便是一个关卡。"如果 x>50 停下来"用LOGO 语句来描述，应表示为："IF（如果）…STOP（停）"形式，则有

IF :x>50 STOP

一旦x的值大于50，海龟便会自动停下来。

x>50 成为停下来的条件，因此我们称此语句为条件语句。

现在我们用条件语句将上面的过程加以改造：

TO LUO :x

```
IF :x>50 STOP
FD :x RT 90
LUO :x+2
END
```

当我们再次键入LUO 5时，螺旋线便会在边长增至50步时，自动地停下来。

通过上面的试验，可以看出当过程调用自己本身时，便会画出奇妙的图形。一个过程被另一个过程调用时，我们称这个被调用的过程为子过程，而把一个过程调用自己本身称为“递归”。

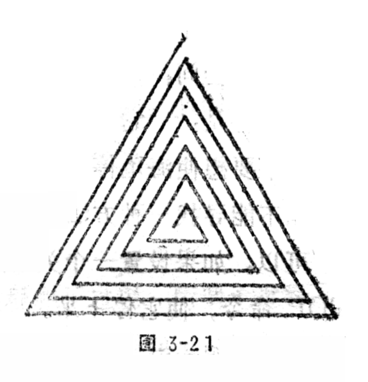


图 3-21

如果将上一过程稍加改动，便能画出更多形状的螺旋线（图3-21、22）。

```
TO LUO1 :x :a
IF :x>50 STOP
FD :x RT :a FD :x
LUO1 :x+3 RT :a
END
```

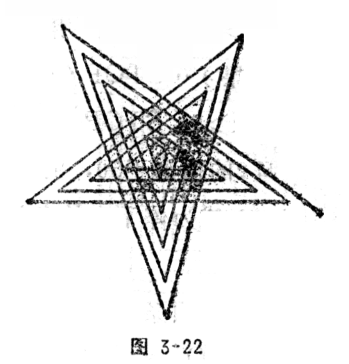


图 3-22

图3-21是x＝5，a＝120；图3-22是x＝5，a＝144。

除此之外，一些简单的过程通过递归调用会引出许

多意想不到的结果。

大家都可以在学习机上作一下这样的实验，先用海龟画一个图形，随便画一个什么图形都可以，然后将它放入递归过程中，看会出现什么结果。

以一个简单的图形（图3-23）为例。

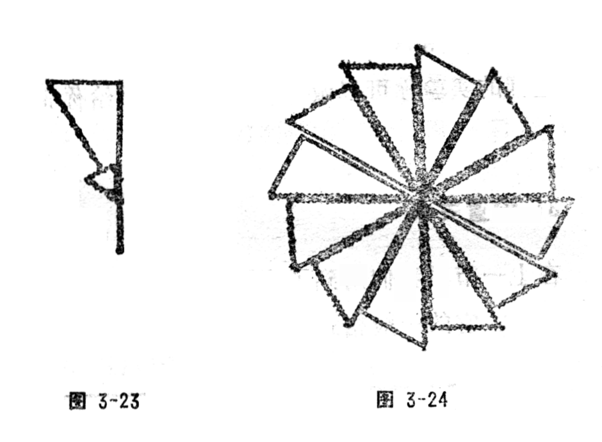


图 3-23　　　　图 3-24

先将图3-23编写成过程，取名 FENGCHE：

```
TO FENGCHE
FD 50
LT 90
FD 20
LT 120
FD 30
```

```
END
```

将它放入递归过程之中

```
TO FENGCHE1
FENGCHE
FENGCHE1
END
```

当我们键入 FENGCHE 1 ↵后，便会出现一个漂亮的“风车”（图3-24）。

这样的实验你可以做许多个，一定会给你带来无穷的乐趣。

## 31.设置关卡

在上一节里我们用到了 IF 语句，IF语句使学习机有了判断能力，它的一般格式：

| IF　条件表达式　　执行某种功能 |
|---|

当条件表达式成立时，就执行方括号里指出的某种功能。

所谓条件表达式就是用“>”；“=”；“<”等连接起来的式子，像上例中的 $x>50$ 或者 $x<3$ 等等。

这里仍将上一节螺旋线的过程作例子。

```
TO LUO :x
```

```
IF :x>50 STOP
FD :x RT 90
LUO :x+2
END
```

首先定义了初始长度为x，进行测试，看是否>50，不大于50，前进x步，右转90度，x增加2，再进行测试，看是否大于50，一旦大于50，便停止执行。由此看来，IF语句起到关卡的作用。但IF语句是不是放在程序的任何地方都能起到关卡的作用呢？这个问题还是试一试再说吧！

我们将IF语句放在程序的倒数第二行，看一看效果如何：

```
TO LUO2 :x
FD :x RT 90
LUO2 :x+2
IF :x>50 STOP
END
```

仔细分析一下，大概学习机永远也不会执行到IF语句。你看，程序执行的顺序一直是前进、右转后调用其自己，又拐回头去执行，关卡起不到任何作用。由此看来，IF语句不是放在任何地方都能起到关卡作用的。

在LOGO语言中，还有另一种设置关卡的方式：

```
TO LUO2 :x
TEST :x>50
IFTRUE STOP
```

```
IFFALSE FD :x RT 90
LUO2 :x+2
END
```

这一过程与前面的过程作用一样，用TEST（测试）来测试x＞50是否为真(TURE)，如果为真（IFTURE），即一旦x＞50，则停止执行。反之，如果为假（IFFALSE)就前进 x，右转90度，继续执行下去，直到x＞50，则停止执行。这与前面的IF语句虽然形式不大一样，但所起的作用是相同的，统称为条件语句。

## 32.图形的生长与中止

当我们将下面过程输入计算机并执行时，会出现什么情景呢？

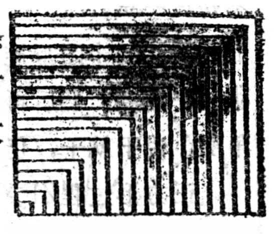
```
TO KUO :x
ZFX :x
KUO :x+3
END
```

其中 ZFX :x 即是我们前面多次出现的画正方形的过程，再重复一遍写在下面：

```
TO ZFX :x
REPEAT 4 [FD :x RT 90]
END
```

当我们键入 KUO 5 后，只见屏幕上一个正方形在不断地扩大生长（图3-25）。

要想使它停止生长，或用CTRL—G 强行停止，或用“IF”语句设置条件；当条件满足时，便会自动地停止扩展。如我们将 IF :x>100 STOP 放入过程的第二行，执行时，当x>100，便会中止生长。


图 3-25

反过来，我们也可以将图形由大到小逐渐缩小，然后人为地控制它，让它在适当的时候停下来。

图3-26有些像一座小塔，是由从大到小一串正方形组成，程序为：

```
TO TA :x
IF :x<3 STOP
ZFX :x
FD :x
TA :x*0.6
END
```

这个图形我们还可以进一步地改进成图3-27。

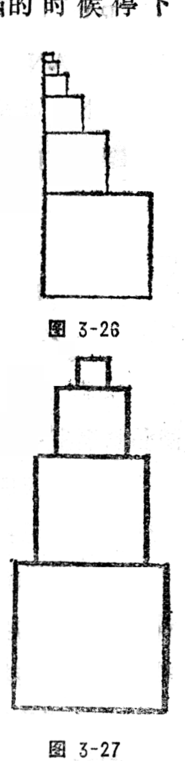

图 3-26

图 3-27

### 33.图形的自旋

当我们学会了让图形不断地生长，然后随意地让它终止以后，又一个大胆的想法产生了，怎样才能让图形旋转起来呢？

不妨试一试。

要让一个图形旋转，就要在画完一个图形之后，让它转动一个角度，然后再画一个同样的图形，再转动一个同样的角度，说干就干，仍用正方形来作试验。

正方形过程ZFX已用过多次，就不再重复了。

```
TO XUAN :a
ZFX 25
RT :a
XUAN :a
END
```

当我们键入XUAN 45 ↵ 和 XUAN 15 ↵，就出现图3-28和图3-29。它们一直旋转着，除非按下CTRL—G键。

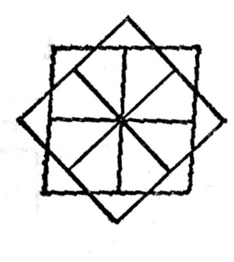


图 3-28

XUAN 45

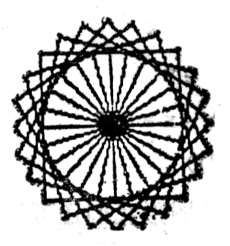


图 3-29

XUAN 15

看来，我们的实验成功了。但我们并不满足，能不能让这些图形一边旋转又一边生长呢？

上一节我们能够让正方形不断地生长，这一节又能让它自旋起来，把这两个程序有机地结合起来，不就既生长又

自旋了吗？如：

```
TO XUAN1 :x
ZFX :x
RT 20
XUAN1 :x+5
END
```

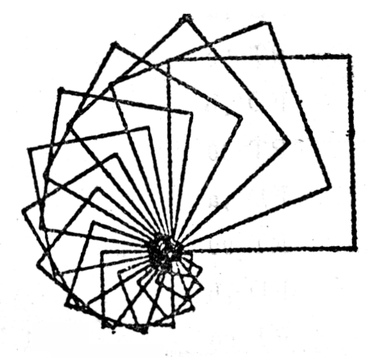


图 3-30

当我们键入 XUAN1 5 ↵后，屏幕上便会出现图3-30。

让我们回过头来分析一下这个过程。先调用正方形(ZFX)过程，画出一个正方形之后，将其旋转20度，遇到递归行，将其边长+5之后，又调用自己再画出以（边长+5）为边长的正方形，又旋转20度。这样将正方形逐渐地扩大并以20度为旋转角，不断地自旋，就会出现图3-30。

### 34.海龟的新作品

通过更深一步海龟课题的探讨，一些令人眼花潦乱的图案不断地展现在屏幕上，使海龟画的图形更加栩栩如生了。在这一部分，我们一直是以正方形作为基本图形来进行变换的，如果改用其它图形，屏幕上会出现什么结果呢？

(1) 以矩形为例，过程名为 CFX

先建立长方形过程，其中变量$a$、$b$分别表示长方形的长和宽。

```
TO CFX :a :b
```

```
FD :a
RT 90
FD :b
RT 90
FD :a
RT 90
FD :b
RT 90
END
```

当键入a、b的数值后，屏幕上会出现长方形，若再以20度为旋转角转动起来，则如图3-31。

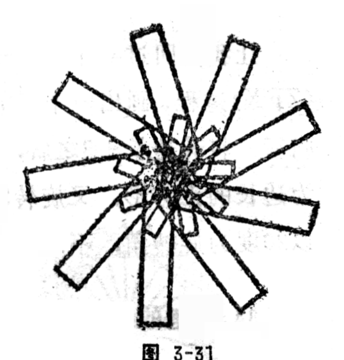


图 3-31

```
TO ZHUAN
CFX 100 20
RT 20
CFX 40 20
RT 20
ZHUAN
END
```

若让长方形一边向左旋转一边不断地缩短，当 $a<10$ 时就要它停下来（图3-32），过程为：

图 3-32

```
TO ZHUAN1 :a
IF :a<10 STOP
```

```
CFX :a 20
LT 30
ZHUAN1 :a-5
END
```

(2) 以三角形为例

先定义三角形(SJX)过程:

```
TO SJX :a
REPEAT 3 [FD :a RT 120]
END
```

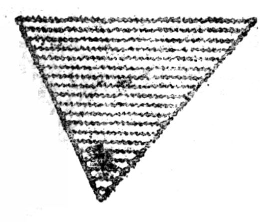


图 3-33

这一次我们首先让三角形不断地生长(图3-33),过程为:

```
TO SJX1 :a
IF :a>100 STOP
SJX :a
SJX1 :a+5
END
```

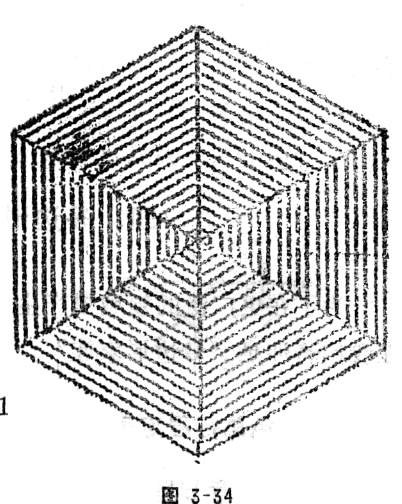


图 3-34

当它边生长边旋转,则得图3-34,即将三角形生长的过程SJX1作为子过程:

```
TO SJX2
REPEAT 6 [SJX1
5 RT 60]
END
```

当 TO SJX3

```
REPEAT  12 [SJX1 1 RT 30]
END
```

则出现图3--35。

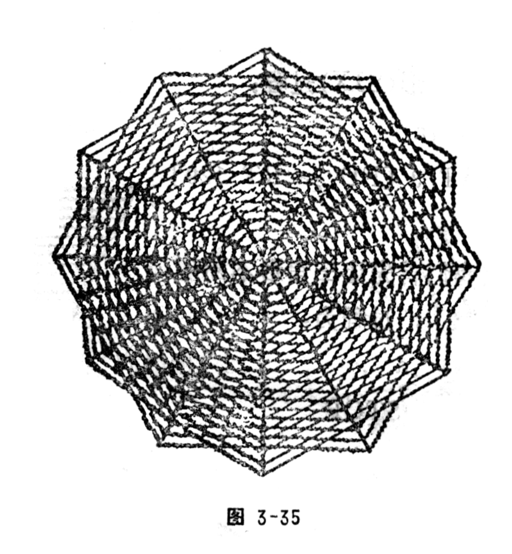


图 3-35

## 四、多边形家族

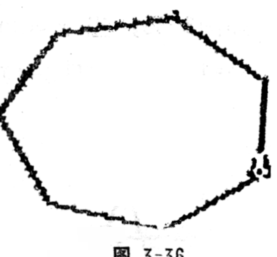
为了在更广泛的领域里产生出更多妙趣横生的图案，现在我们将涉足“多边形家族”。

### 35.海龟总行程定理

在前面的章节里，我们已经接触过多边形。如画正五边

形在本章“一、23.”中做过。

画正七边形（图3-36，a=40）：


图 3-36

```
TO QBX :a
REPEAT 7 [FD :a LT 360/
7]
END
```

还可以接着画下去，如正八边形、正九边形、正十边形。

但不管海龟走过什么路程，要画出多边形来，必然要转回到起始点，即旋转360度或者360度的倍数，这就是海龟总行程定理。根据这个定理，我们写出正多边形过程，其中 x 代表边长，a代表角度：

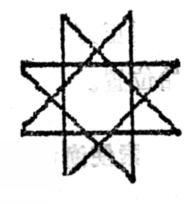
```
TO ZDBX :x :a
REPEAT 360/a [FD :x RT :a]
END
```

这样，当我们给出不同的角度时，就会画出不同的正多边形，即给出90度画出正方形，给出36度，画出正十边形。

如果我们要画出任意多边形，过程又如何写呢？

我们可以引用递归来完成：

```
TO DBX :x :a
FD :x
RT :a
DBX :x :a
END
```


图 3-37

当 x=50，a=135 时，如图3-37；当x=50，a=144时，

如图3-38；当 x＝50， a＝170时， 如图3-39； 当 x＝60，a＝150时，如图3-40。



图 3-38　　图 3-39　　图 3-40

看这些多边形家庭的成员个个神态各异，独具风格。但你千万不要忘记，这个过程在你未键入 CTRL-G 键时，它还一直在运行！

你可以尽情地用这个过程去作更多有趣的实验，输入不同的数据：可以是非常小的一组数，像DBX 10 1 也可以是极大的一组数，像 DBX 200 1000 等等，看屏幕上将会出现什么情景。你也可以做一种游戏：先想一下你将要送进去的一组数会形成几边形，然后看一下屏幕效果与你猜想的是否相符，看看边数和角度之间是否符合海龟总行程定理，即海龟走完全程正好要旋转 360 度的整倍数，并且海龟的指向仍回到初始位置。

## 36.探索使海龟停下来的新方法

上一节里我们已经探讨了画出任意边数多边形的过程。

TO DBX :x :a

```
FD :x
RT :a
DBX :x :a
END
```

这是一个递归过程，如果不加条件限制的话，它会无休止地运行下去，怎样才能让它自动停下来呢？前面我们已经习惯了用 IF 语句设置关卡，使海龟在条件满足时停下来，如 IF：边长>100 STOP。

那么要使海龟画出一个多边形后能够准确地、自动地停下来，又要设置一个什么条件才合适呢？

这时我们想到了“海龟总行程定理”——海龟从起始点出发，旋转一圈之后，又回到原地，且海龟指向与起始时相同。那么我们就要查看海龟的朝向在旋转一周之后是否与起始方向相同即可。前面我们已经接触到 HEADING 命令，这个命令的作用是检查海龟的朝向。假若海龟的起始位置为竖直向上（即为0），那么我们就以 HEADING=0 作为条件来试一试：

```
TO DBX :x :a
FD :x
RT :a
IF  HEADING=0 STOP
DBX :x :a
END
```

当我们的海龟以竖直向上的朝向作为画多边形的起始朝

向时，走完全程即画完一个完整多边形，便会自动地停在预定位置。

但假如海龟不是在 HEADING=0 时开始作图的，那么海龟仍然停留在 HEADING=0 处，这样所画的多边形就不会是完整的。怎样来改变这种状况呢？那就要重新设置使海龟停下来的条件——让 HEADING= 起始位置。现将起始位置起名为A，过程将修改成：

```
TO DBX :x :a
FD :x
RT :a
IF HEADING= :A STOP
DBX :x :a
END
```

那么究竟A是个什么数值呢？电脑无从知道。但我们可以给A一个具体的值，用 HEADING 命令测试海龟的初值位置，然后把这个位置参数赋值给A，具体如何去赋，这要用到LOGO语言中的命名命令：MAKE。

命名是LOGO语言中一个非常重要的组成部分。我们可以给过程命名，如让海龟画一个正方形，我们给它取名为ZFX；可以给参数命名，如为了画出不同边长的正方形我们给过程ZFX定义边长为x。除此之外，我们还可以为程序中各种数据命名，如将20命名为A，在计算机里，每一个变量如同一个小匣子，把20命名为A，实际上是把20放进名字为A的小匣子里面，用 MAKE"A 20这条命令便可实现。

因此，命名命令MAKE的一般形式为：

| MAKE ″变量 表达式 |
|---|

功能是将表达式的值赋给变量，或者将表达式的值命名为变量，并且通过在名字前加符号“″”来表示。

MAKE 也可以为表达式命名，从而实现运算功能，如：

```
MAKE ″ A 4*(3+2)/(3+2)+1
PRINT :A
```

屏幕上显示出：

```
5
```

这里 PRINT :A，是输出A的值；显然，现在A的值即为表达式的计算结果。

现在我们就可以解决前面提出的问题了。先将多边形过程列出：

```
TO DBX :x :a
IF HEADING= :A STOP
FD :x
RT :a
DBX :x :a
END
```

这里使用了新的变量A。取海龟的当前位置为A的值，而A的值代表海龟原来的起始位置。但是我们并没有把

这一事实告诉学习机，因而当我们执行这段过程，即键入DBX 50 90 时，学习机就会发出出错信息，意思是没有给出A的值。

现在学习了MAKE 命名命令，就有办法将海龟的原始朝向送给A了。

MAKE "A HEADING

HEADING 表示了海龟的起始位置，为了避免每次运行时都要键入以上命令行，我们最好将它写入过程中，现在我们将画多边形过程取名为DBX1：

```
DBX1 :x :a
MAKE "A HEADING
DBX :x :a
END
```

我们把这个过程分析一下，先将海龟的起始方向HEADING 命名为A，然后调用子过程 DBX，这个子过程就是我们刚才叙述过的过程：

```
TO DBX :x :a
IF HEADING= :A STOP
DBX :x :a
END
```

这样，每当画出一条边就检查当前的方向是否等于A的值，一旦等于A的值，即起始方向，过程便停止执行。

我们可以从在学习机上的实际操作中得知，不论海龟原来在什么位置，从什么方向开始画多边形，都能够在画出一

个完整的图形后准确地停下来。

### 37.多边形的螺旋线

螺旋线和多边形有亲缘关系。螺旋形可以看成为生长的多边形。其过程如下：

```
TO DBXL :x :a
FD :x
RT :a
DBXL :x+1 :a
END
```

取 x＝1，a＝45 得图3-41；取x＝1，a＝90 得图3-42。

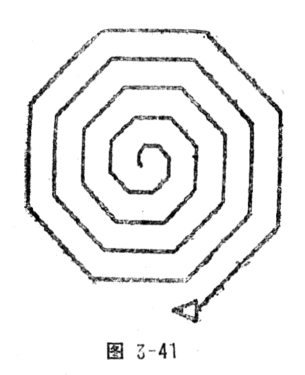


图 3-41

图 3-42

我们还可以以各种不同的角度作为输入数来运行DBXL，当螺旋线变得非常大时，再按CTRL-G 键，使它停止。

当然，我们还可以对这个过程作进一步的修改，使它的边长每次的增量可以改变。我们用变量来作为增量：

```
TO DBXL1 :x :a :y
IF :x>70 STOP
FD :x
RT :a
DBX1  :x+:y :a :y
END
```

取 x＝1，a＝45，y＝1 得图 3-43；取 x＝1，a＝45，y＝3 得图3-44。

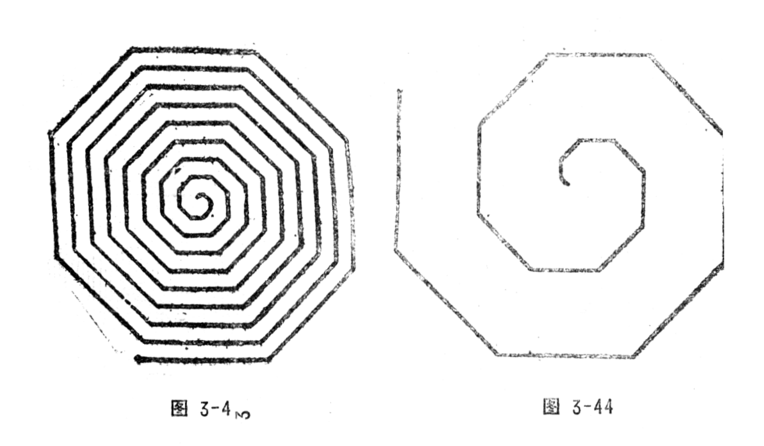


图 3-43　　　　　图 3-44

## 38.多边形的内螺线

多边形的内螺线的形成，是靠角度的变化，即使多边形的每个角度数不断地增加或者减少。过程如下：

```
TO DBXN :x :a :y
FD :x
```

```
RT：a
DBXN ：x  :a+:y  :y
END
```

这里我们用y作为角度的增量。

当我们在电脑上去执行此过程时，出现的情景简直使你大吃一惊！如图3-45～49。

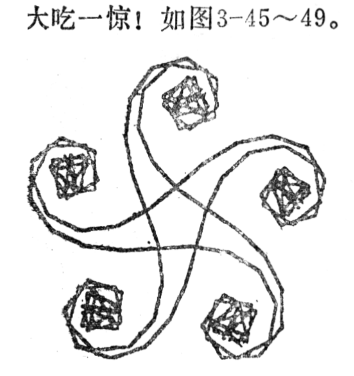


图 3-45
DBXN 10 1 10

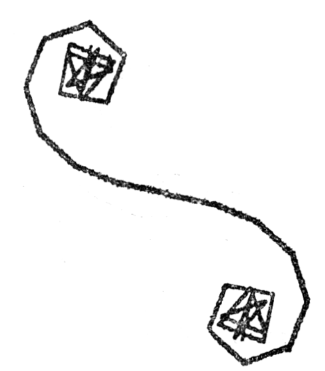


图 3-46
DBXN 10 10 10



图 3-47
DBXN 7 3 10

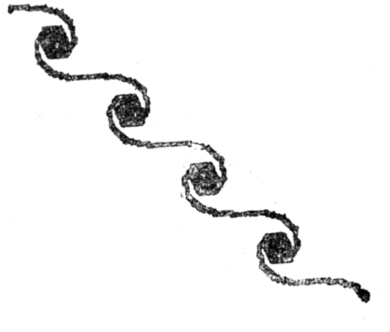


图 3-48
DBXN 7 5 10



图 3-49

DBXN 20 10 7

## 39.多边形的家属



图 3-50

2DBX 30 60 60 210

多边形是一个大的家族，在这个大家族中每个“人”的形态各异。这一节里，我们再例举一些。

多边形的变型很多，如图3-50、51、52。其过程为：

```
TO 2DBX :x1 :a1 :x2 :a2
FD :x1
RT :a1
FD :x2
RT :a2
```

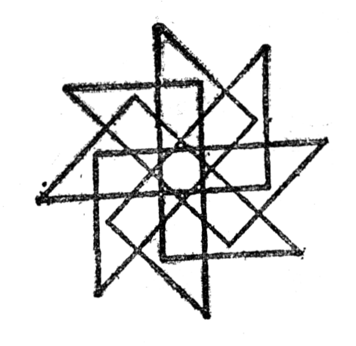


图 3-51
2DBX 30 90 50 135

图 3-52
2CBX 25 190 50 200

2DBX :$x_1$ :$a_1$ :$x_2$ :$a_2$

END

这里我们采用了四个输入参数，使图形又有了一些新的变化。

而图3-53、54是如下过程运行的效果：

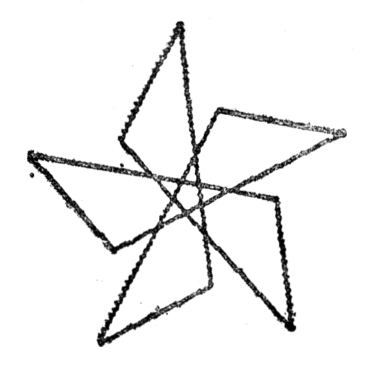


图 3-53

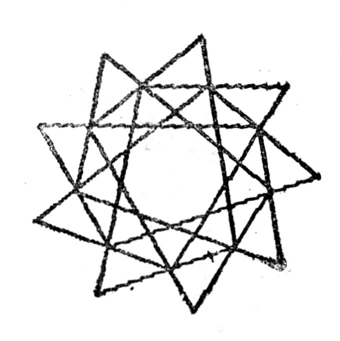


图 3-54

```
TO 3DBX :x :a
FD :x
RT :a
IF :HEADING=0 STOP
3DBX :a :x
END
```

注意在最后一行中两个变量交换了位置。

图3-55～58也都是多边形家族的成员，你能想象出它们的画法吗？

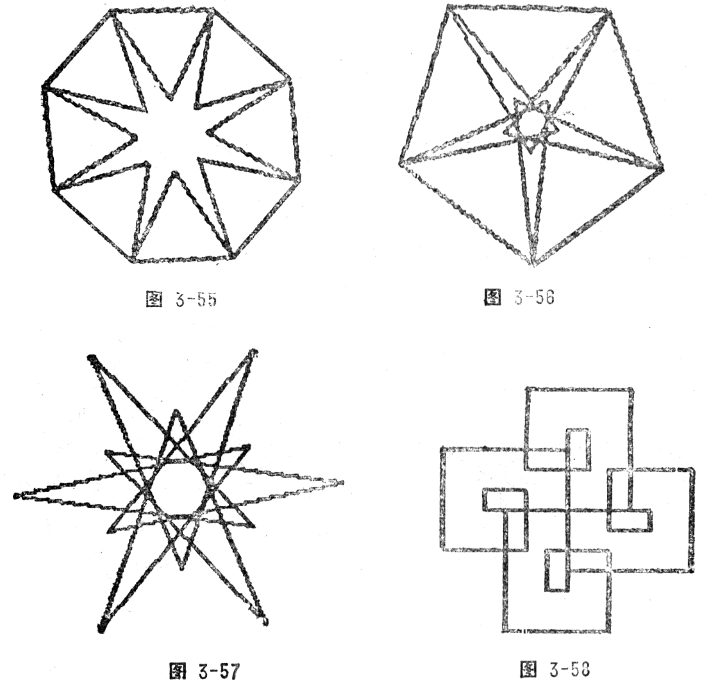


图 3-55　图 3-56

图 3-57　图 3-58

## 五、圆和弧组成的图案

前面我们是以正方形、三角形等多边形作为例子绘制了形态各异的图形，现在我们要让海龟在圆和弧的领地里得到充分地发展。

### 40.新“部件”——圆和弧

圆和弧在绘图中占有很重要的位置，但由于画起来比较麻烦，因而LOGO专门定义了绘制圆和弧的过程；它们存放在磁盘里，需要时，从磁盘调入内存，方法如下：

键入：READ ″ CIRCLES

若你的磁盘上确实有圆和弧的过程，在回车后，便会听到驱动器“呼呼”作响，几秒钟后，屏幕上会出现有关圆的过程名：RCIRCLE 和 LCIRCLE 用来画圆，R A R C 和LARC画圆的 1/4；每一个命令后须跟一个输入数——要画的圆或弧的半径，如图3-59是不同半径的圆，图3-60是不同半径的圆和弧。

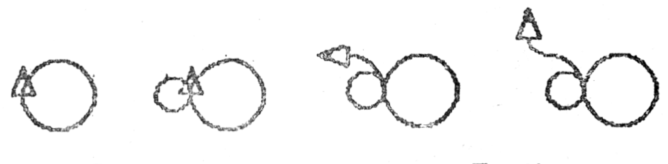


图 3-59　　　　图 3-60

如果你的磁盘上没有圆和弧过程，也不要因此而犯愁，

自力更生写一个圆的过程。

我们根据让海龟每前进一小段固定的距离，旋转一个固定的小角度，一遍一遍反复地进行，当海龟转够 360 度时，所形成的轨迹，就成为一个圆。过程为：

```
TO RCIRCLE
REPEAT 360 [FD 1 RT 1]
END
```

画弧的过程实际上是圆过程的一部分，我们用 $y$ 来代表要画弧的度数，则

```
TO RARC :x  :y
REPEAT :y [FD :x RT 1]
END
```

当我们键入 RARC 1 90 ↲时，便画出右四分之一圆。

以上过程虽然能够画出圆或弧，但是进行的速度太慢，每次只能前进一步，旋转一度，实在太麻烦了，因为在显示屏的精度范围内，一个具有20条边的有规则的多边形与一个圆已很难区分了。

我们可以用画出36边形的过程来代替圆过程：

```
TO RCIRCLE :x
REPEAT 36 [FD :x RT 10]
END
```

我们将 x 定为10，这样运行起来就快了10倍，其效果相差无几；若把海龟隐藏起来，速度会更快（图3-61）。

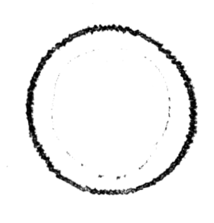


图 3-61

## 41.圆和弧组成的图案

学习机已会画圆和弧了，现在可以用圆和弧来组成有趣的图案了，图3-62是一朵美丽的花，花瓣是由两条弧（图3-63）组成的。其中画花瓣的过程如下：

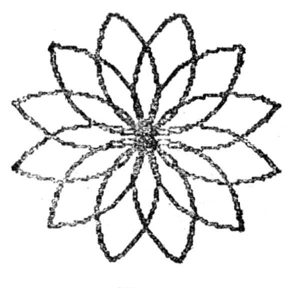


图 3-62

```
TO HUAB :x
RARC :x 60
RT 120
RARC :x 60
RT 120
END
```



图 3-63

从过程我们可以看到要画花瓣，必须在画完一条弧后转动海龟，再画出同样的一条弧来。这些参数大家可以在电脑上试一试，大概可画出许多胖瘦不一的花瓣来。

现在我们要用六个同样的花瓣就可以组成一朵花（图3-64）：

```
TO HUA :x
REPEAT 6 [HUAB :x RT 60]
END
```

当然，也可以根据自己的爱好画出更多的花瓣，组成更

美丽的花朵。同时，如果有兴趣的话，还可以给它加上枝、叶，成为一枝花，进而组成一束花或一个花园（图3-65）。

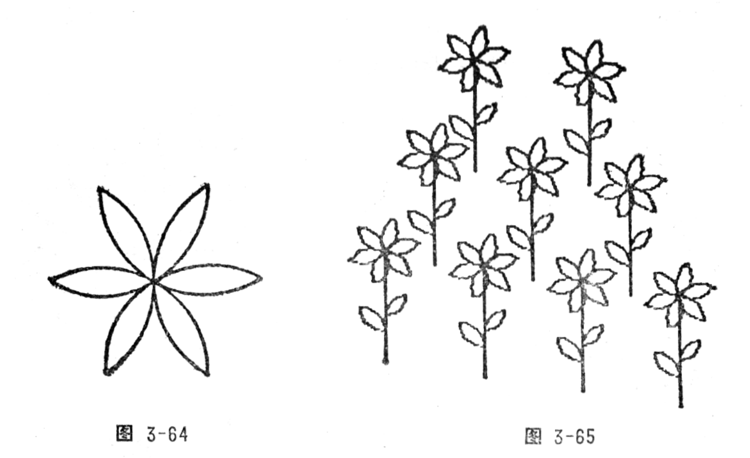


图 3-64　　图 3-65

图3-66是一个太阳的图案，它是由一条条“蛇”形曲线（图3-67）组成的。

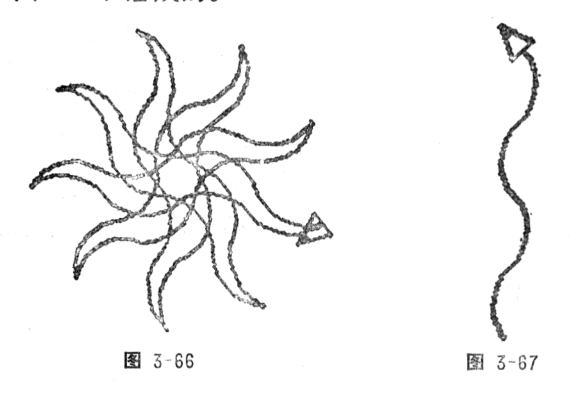


图 3-66　　图 3-67

“蛇”是由左、右两弧相接而成的，其中LARC :x :y 为左弧过程。

```
TO SHE :x
LARC :x 90
RARC :x 90
LARC :x 90
RARC :x 90
END
```

当我们把这一条条“蛇”，旋转一定角度，便会得到漂亮的太阳图案（图3-66）。

```
TO TY :x
REPEAT 9 [SHE :x RT 160]
END
```

图3-68、69也是由弧组成的。

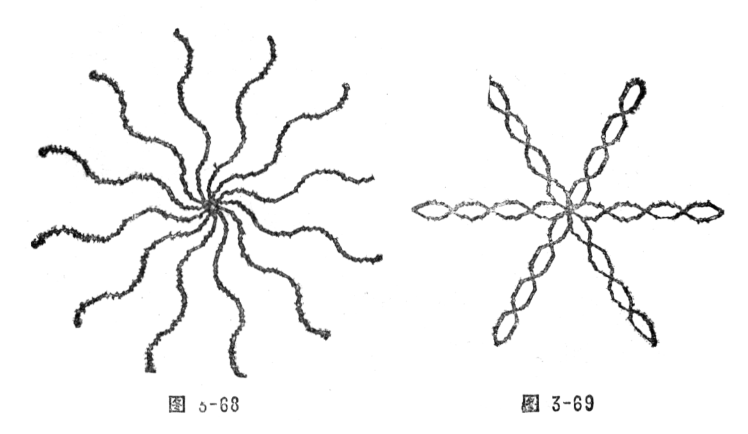


图 3-68　　　图 3-69

## 42.圆和弧的自旋

圆和弧也同多边形一样，可以自旋，而且圆和弧自旋所

绘之图更美。图3-70、71、72就是由圆过程经过旋转而得

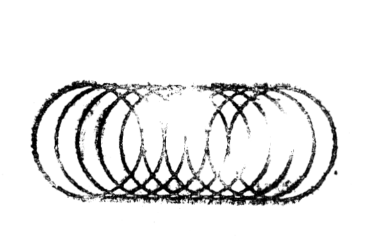


图 3-70

图 3-71

到的。当然也要用到递归技术。其中图3-72的过程就是海龟每画出一个圆之后，让它转过20度，再画一个圆，因而将过程取名为 UAZ :x，过程如下：

```
TO UAZ :x
RCIRCLE :x
RT 20
UAZ :x
END
```

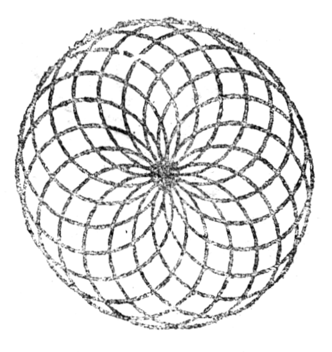


图 3-72

图3-70则是海龟每画出一个圆之后，让它前进一小段距离，再画下一个圆。其过程如下：



```
TO UAZ1 :x
RCIRCLE :x
PU RT 90 FD 10 LT 90 PD
UAZ1 :x
```

```
END
```

我们还可以列举出将递归技术运用在作圆和弧组成的图案上，产生许多有趣的图案的例子，比如我们可以画一双猫眼睛（图3-73）。它是由两个对称图形组成，先画出其中的一只，然后转动180度产生出另一只。过程为：

```
TO YAN1 :a
IF :a>8 STOP
CIRCLER :a
YAN1 :a+1
END
```

图 3-73

其中圆过程CIRCLER :a如下：

```
TO CIRCLER :a
REPEAT 360/ :a [FD 2 RT :a]
END
```

当我们键入：YAN1 2 ↵，便会在屏幕上，画出其中的一只猫眼，然后用它组成一双猫眼：

```
TO YAN2
YAN1 2
RT 180
YAN 2
END
```

圆和弧通过自旋，或者其它组合还可以产生许多奇妙的画案来（图3-74、75、76、77）。

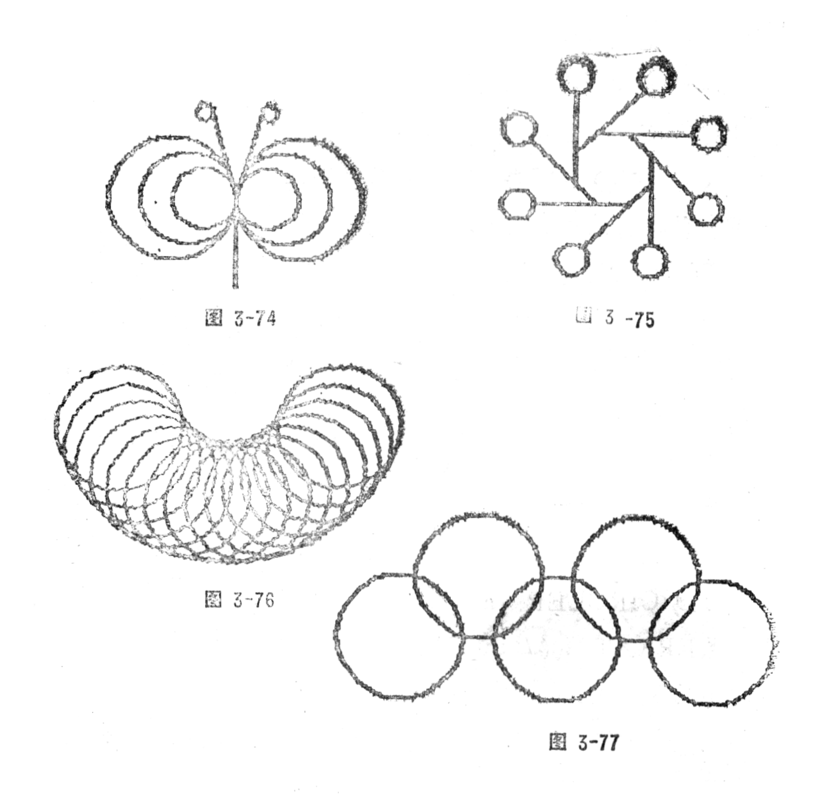


图 3-74

图 3-75

图 3-76

图 3-77

## 六、瞬息海龟和动态海龟

LOGO语言被誉为“电脑教室里的国王”，可以开发儿童的智力，培养孩子的想象力。那么几岁的儿童可以开始学习LOGO语言呢？回答是：幼儿园五六岁的小朋友就可以坐在计算机旁做一些适合他们年龄特征的题目，这就需要家长

和教师去创造这样的学习环境。

本章以“瞬息海龟”作为标题向年幼的孩子介绍一种新的方式，即不需要学会LOGO命令、输入数等概念性的知识就可以直接运用海龟在电视机屏幕上做图，发挥自己的想象。同时也向家长和教师提供此类程序的原理和编程方法。另外以“动态海龟”为标题给孩子们创造一个新的电脑天地——电子游戏。通过饶有风趣的游戏使孩子们爱上电脑，并在教师和家长的指导下能不断地创造新的游戏，改进完善旧的游戏，使电脑的神秘感从孩子们的脑海里消失。

## 43.瞬息海龟

当年幼的儿童初次接触电脑时，要让他们搞清楚命令、语句等概念将是非常困难的，而如果让他们利用键盘上指定好的几个键来直接指挥海龟去做图，却是轻而易举就可以办到的事情，我们就把这种工作方式称之为瞬息海龟，给以下几个单键赋于新的内容：

[F] 键：指挥海龟向前进10步；

[R] 键：指挥海龟向右转过30度；

[L] 键：指挥海龟向左转过30度；

[C] 键：清除屏幕。

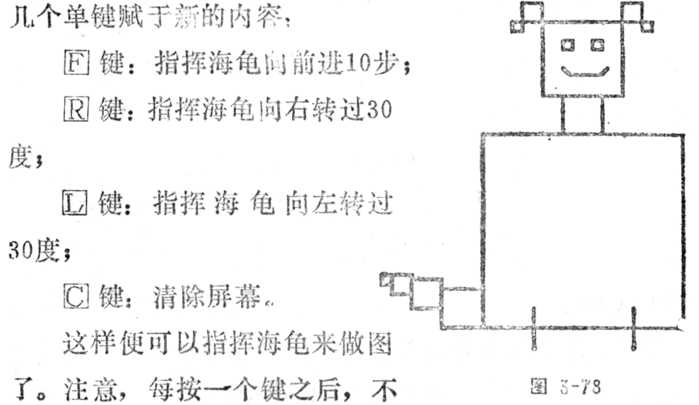


图 3-78

这样便可以指挥海龟来做图了。注意，每按一个键之后，不

必按回车键。做完一幅图之后，按C键可以将屏幕清除干净，准备作新图。

用“瞬息海龟”作为工具，可以在屏幕上画出任何图画。图3-78、79、80便是小朋友们用“瞬息海龟”绘制出的作品。

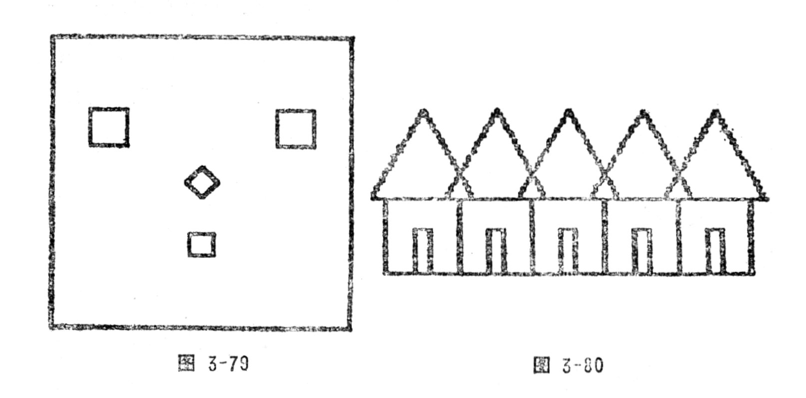


图 3-79　　　　图 3-80

## 44. 用“瞬息海龟”作图的方法

在这一节里要向家长和教师介绍如何建立瞬息海龟作图方式，去创造一个适合自己的孩子和学生学习电脑的环境。

上一节里我们介绍了所谓瞬息海龟作图，就是靠键盘上单键输入，每一个键代表一个动作。但这些都是我们人为规定的。我们之所以这样规定是选用了 FORWARD(FD)、RIGHT(RT)、LEFT(LT) 的字头。当然还可以定义更多的功能键。那么如何让计算机知道某一个键按下去后要执行什么功能呢？这就是瞬息海龟作图的关键所在。要解决这个

问题，就要求助于LOGO的一个新命令：READCHARACTER命令，此命令称之为键盘输入命令：

```
READCHARACTER
```

它的作用是等待你从键盘上键入一个字符，当计算机执行到 READCHARACTER 语句时便会暂停等待你从键盘上键入一个字符，不需紧跟 RETURN（回车）接着便可以根据你的程序安排向下执行。

下面我们便可以给出瞬息海龟作图的程序了，我们将此过程取名为 SCHG：

```
TO SCHG
ML
SCHG
END
```

其中 ML 为子过程，其内容如下：

```
TO ML
MAKE ″x READCHARACTER
IF :x= ″F FD 10
IF :x= ″R RT 30
IF :x= ″L LT 30
IF :x= ″C CS
END
```

这就是使瞬息海龟作图的过程，它的结构非常简单，首

先调用了ML这个子过程；然后再调用其自身。

其中过程ML的作用是从键盘接收一个单键之后，转去执行此单键的规定动作。

过程的第一行是给按下的任意一键取名为$x$，接收键盘一单键赋于$x$。然后检查它是否是F、R、L、C键，若是F键，则命令海龟前进10步；若是R键，则命令海龟向右转30度；若是L键，则命令海龟向左转30度；若是C键，则清除屏幕。若按这四个键之外的任意键，则屏幕上不会有任何反应。

当然，我们还可以定义更多的键让海龟去执行更加完善的动作，那就是在过程ML中增加相应的附加条件便可以实现，如你可以用U键命令海龟抬笔，用D键命令海龟落笔，用Z键指挥海龟画出一个小正方形等等。当然要利用我们前面章节里已定义过的正方形过程ZFX。

以下三行程序加在ML过程中END前，这些功能就可实现了：

```
IF :x= "U PU
IF :x= "D PD
IF :x= "Z ZFX
```

孩子们在“玩海龟”过程中的任何一个想法，任何一个新的建议，家长和教师都不要放过，尽量地满足他们，不断地完善、修改“瞬息海龟”过程，只有这样，孩子们的创造力才会得到充分发挥。

最后让我们拐回头来总结一下瞬息海龟的作图方法：

①首先家长或教师根据孩子们的要求建立SCHG过程。

②键入 SCHG ↵ 将瞬息海龟过程调出，给孩子们讲清规定好的各个键的功能，指导孩子在屏幕上用这些键去进行创作。

③在创作过程中，家长和教师要不断地发现和总结SCHG过程不够完善的地方，尽量地去修改此过程。

## 45.动态海龟

孩子们大概都十分喜爱电影或电视节目中的动画片，如《大灰狼和小白兔》、《米老鼠和唐老鸭》等，如果让海龟作为演员，我们的孩子作为导演，自己去创作动画片，那一定会给孩子们带来更大的乐趣。

要创作动画片，就要想办法使海龟像有生命的小动物一样动起来，而我们的小导演可以控制它运动的方向和运动速度，还可以控制屏幕的颜色，当然也可以别出心裁，让小海龟突然跳起到屏幕的某一位置。若我们的小导演认为海龟累了，也可以让它休息片刻再继续活动。

我们也可以让动画片加进一些游戏的内容，比如让海龟绕着圆形跑道运动而不出界，等等。这些都是属于动态海龟的范畴。动态海龟的出现，又会给孩子们的海龟课题增加更丰富多彩的内容。

## 46.用动态海龟作图的方法

在“瞬息海龟”中，我们掌握了如何使用键盘上的键来

控制海龟，如按F键，使海龟前进10步。那么，要使海龟运动起来，就要使海龟不间断地运动。因此，我们首先要编制一个过程，能使海龟不断地运动，我们给它取名为运动YUD，在这个过程中，我们要给出运动的速度 $v$ 。过程如下：

```
TO YUD :V
FD :V
YUD :V
END
```

执行这个过程就使海龟不间断地前进。YUD 10 ↵表示以每次10步的速度迈大步"跑"来。而 YUD 5 ↵ 表示放慢速度"走"。这样我们的"动画片"完成了第一步——使"演员"海龟动了起来，下一步就要实现如何在导演的控制下完成各种动作。

这就要求助于"瞬息海龟"中，ML 子过程了。在瞬息海龟过程的 ML 子过程中，我们用到了 READCHARACTER 命令，它的作用是等待键盘输入一个字符。现在我们要使海龟不间断运行而能接受控制，就是说如果从键盘上键入一个字符使海龟对表示的内容作出响应，而如果什么也不键入，海龟仍然能维持原来的动作。在LOGO语言中，要用RC?命令来实现：

| RC ? |
|---|

它的作用是根据是否已键入字符分别输出 TRUE 和

FALSE。

现在我们可以写出动态海龟的子过程 ML：

```
TO ML
MAKE "x READKEY
IF :x= "R RT 30
IF :x= "L LT 30
END
```

在程序的第一行中调用了 READKEY，这是一个过程，它的作用是如果键入一个字符，则输出的就是那个字符，否则什么字也不输出。

```
TO READKEY
IF RC ? OUTPUT READCHARACTER
OUTPUT"
END
```

将子过程ML加入动态海龟过程YUD :V中，便可以在运动中控制海龟的行动了：

```
TO YUD :V
FD :V
ML
YUD :V
END
```

当然，事先要建立起子过程ML和ML的子过程READ-KEY。

现在我们的动画“演员”海龟可以在小导演的控制之下

自由行动了，但总觉得这样的动画片似乎有些过于呆板，演员仅仅会前进和向左向右转，能不能再生动一些呢？比如你想让海龟前进的速度越来越快，或者相反，又比如你想根据自己的愿望来改变画笔的颜色，等等。这些功能的实现，要通过改变ML过程来完成：

我们先写出改变海龟前进速度的程序行：

```
IF :x= "A MAKE V :V+1
IF :x= "B MAKE V :V-1
```

这样当我们键入A或B键后，海龟便会以每次递增一步或者递减一步的速度来运行。

改变画笔颜色的办法之一是使用数：

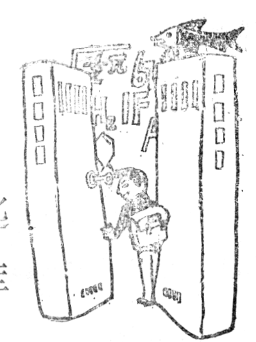
```
IF :x=0 PC 0
IF :x=1 PC 1
```

当键入数字0键之后，画笔为黑色，当键入1键，画笔变为绿色。当然我们还可以设计其它的颜色。

通过不断完善ML过程，动画片就会越来越生动了。

最后我们还可以用动态海龟设计一些游戏。如我们可以在屏幕上画一个环形“跑道”，让孩子们驱赶动态海龟绕跑道一圈，以撞跑道边的次数最少为胜，那将更加有趣。

# 第四章 用智慧的钥匙打开知识宝库



这一章我们准备从解决实际问题入手，按知识的深浅分阶段，深入浅出地介绍BASIC语言及编程技巧。

BASIC 语言是众多程序设计语言中运用比较普遍的一种，适合初学者使用。其一，它简单，并且使用范围广；目前大部分教学辅助软件都是用 BASIC 语言编写；其二，它直观性强，具有计算机与人对话的功能——有问有答，使用方便。

学会使用 BASIC 语言编写程序，不但可以解决身边的实际问题，而且可以从中尽兴地发挥个人的主观能动性和创造力。编写一个较为复杂的程序，一百个人如果分别独立进行程序设计，编出的程序虽然都达到解决问题的目的，却可能没有某两个程序是完全相同的，这说明由于每个人的思路不同，也说明程序设计本身存在着灵活性和创造性，从而使个人的才智得以发挥。另一方面，编制程序的本身就是进行着结构性思维和系统性思维的锻炼，从而可以培养科学性的思维

方法，还是让我们拿起这把智慧的钥匙，尽快打开知识宝库的大门。

## 一、初步的程序设计

下面我们通过解决几个实际问题，掌握 BASIC 的基本语句，学会初步的程序设计。

### 47.一个程序的执行过程

（1）先把内存贮器清理干净

就像我们每次写字之前要把黑板擦干净一样，在向学习机送入程序之前，也必须把内存贮器清除干净，否则，原来内存贮器中的旧程序和新键入的程序混杂在一起，计算机执行起来就会乱套。

所以，我们必须养成一个习惯，在送入一个新的程序之前，首先键入：NEW 命令

方法是依次按N、E、W三个英文字母，再按RETURN（回车键）。

这时内存中的东西就被清除干净了，但屏幕上并没有什么反映——对程序的执行没有什么影响。如果你也想把屏幕擦得干干净净，就请键入：

HOME ↵

这时屏幕上就什么也没有了，只有一个光标在左上角闪烁，告诉你现在可以输入程序了。

(2) 向学习机送一个程序

下面让我们一起来将一段程序送入学习机。有空格的地方一定要按键盘下方最长的那个键——空格键，每一行结束都要键入 [RETURN] 键。

```
10 PRINT "A";
20 PRINT "B";
30 PRINT "C"
40 END
```

在这个程序中我们可以看出，一个程序是由若干条语句组成的，每一条语句的第一段为号码，称为语句标号，它必须是正整数，它告诉学习机执行的顺序。在正常情况下，学习机总是从语句标号小的向语句标号大的方向执行，相邻语句行之间语句标号要留有一定的间隔，这是为了在修改程序时插入新的语句。

除语句标号外，其余部分为语句体，它包括语句定义符和语句体。语句定义符是规定计算机执行某一特定的功能，如10号语句中的 PRINT，就称为语句定义符，它的功能是要学习机屏幕显示结果或者是将结果送打印机。

语句体是规定语句定义符要执行的具体内容，上面程序第10行语句中，"A"，就为语句体，它前面的语句定义符PRINT 是要求将结果送屏幕显示，即显示字符A。

上面我们每键入一行程序后按回车键，就将这行程序送入内存，当最后一个"回车"键入之后，程序便全部进入内存。

(3) 列程序清单和修改程序

①若要检查内存现有的程序，可键入：

LIST ↵

这时屏幕上会显示出学习机内的全部程序。如果只需要显示20到40号语句之间的部分程序清单，可键入命令 LIST 20，40 ↵，这时屏幕上显示20到40号语句的内容。

②修改程序：

A.在输入程序过程中修改：在一个程序输入过程中还未按回车键之前，发现有错字符时，可用◁键（退格键）来修改。它如同橡皮一样，每按一次键，就擦去一个字符。

B.输入内存后程序的修改：若送完一行程序并按过回车键时发现有错误，可以重新将原语句行按正确的键入一遍，则同一行号原有内容便被新键入的内容所代替。也可以用键盘编辑键进行修改，具体方法参看第五章。若要删除一行程序，可按该行程序编号再回车，该行即被删除。

（4）运行程序

程序键入内存后并不意味着它会自动运行，必须在程序检查无误后键入：

RUN ↵

程序便开始运行。如果是我们刚才键入的程序，则屏幕上便出现：

ABC

这便是我们上面程序的执行结果。

若程序有错误，在执行过程中会自动打出错误信息，在改正错误之后，才能重新运行。

## 48. 让学习机来为我们做题

### ——介绍赋值语句LET

在让学习机做题之前，必须要把一些概念和计算机语言中的规定搞清。

（1）内存单元、变量和数据

计算机的内存就像许许多多记忆的小盒子一样，这一个个小盒子就叫做内存单元。要想让学习机做题，必须先把数据存入内存单元，这时我们就要给这个内存单元起个名字，叫变量名。单个变量叫简单变量，它的名字可以由一串字母和数字组成，但是第一个字符必须是字母，比如：A、B2等等。

变量又分为数值变量和字符串变量。上面举的两个变量名就为数值变量名，数值变量可以参加算术运算、逻辑运算和关系运算，它的值是具体的数据，而字符串变量的值是一串字符，它主要用于文字描述。字符串变量名由数值变量名之后加一个“$”符号组成。如：A$，B1$等。

（2）给变量赋值——LET 语句

可直接给变量赋值的语句 LET：要想让学习机做题，首先应该把数据送入学习机，也就是要先给变量赋值：

```
10 LET A=3
20 LET A$= "ABC"
30 PRINT A
40 PRINT A$
```

```
50 END
```

键入 RUN ↵

屏幕显示

```
 3
ABC
```

这就表明A单元里装入3，A$单元里装入ABC。

在10号语句中，LET是语句定义符，表示给变量赋值。A=3 为语句体，表示将3送入变量为A的单元里。

内存装入数据有这样的习性：当你需要数据时可以随时取出。如将A中的数据3取出，它仅仅是复制品，名字为A的小盒子里仍然保留着3，除非有新的数据装入，才能将3覆盖掉。

再看下面的程序，

```
10 LET A=3
20 LET B=A
30 LET A=5
40 PRINT A, B
RUN ↵
5       3
```

上面程序中，先将3赋给变量A，再将A的值赋于B，因为A中仍保留3，所以B的值为3，又将5赋于A，这时A里的3就被5代替，因此A的值就变为5。

LET A=3 并不表示A等于3。如 A=A+1，这在数学里是讲不通的，而在 BASIC 语言里却有实际的意义，

表示将A这个单元的内容加上1之后，再存入A单元。例如，刚才A单元里为3，那么将3取出之后再加上1送入A，现在A就变成为4。

总结一下 LET 语句的格式：

| 行号 LET 变量=表达式 |
|---|

意思是将等号右边的表达式的结果赋值给等号左边的变量。

(3) BASIC 语言中的算术表达式

BASIC 语言中的算术表达式与数学中的略有不同，列于表4-1。

表 4-1

| 数学表达式举例 | BASIC 表达式举例 |
|---|---|
| $a+b$ | $A+B$ |
| $a-b$ | $A-B$ |
| $ab$ | $A*B$ |
| $a\div b$ | $A/B$ |
| $a^3$ | $A\wedge 3$ |
| $\frac{a+b}{3c}$ | $(A+B)/(3*C)$ |
| $5[(a-5)(a+b)]$ | $5*((A-5)*(A+B))$ |

由表4-1可以看出，BASIC表达式中只准用圆括号，乘

号不能省略，一律要用“*”，乘方要用“∧”。

（4）让学习机为我们做题

一切工作准备就绪，下面开始让学习机为我们做题了。

题目是：求圆管的体积（图4-1）。其中 $a=6$ 厘米，$b=5$厘米，$c=100$厘米；

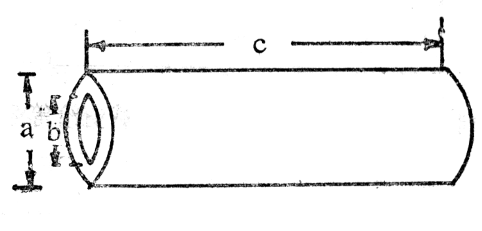



图 4-1

体积公式为：$V=\frac{\pi}{4}(a^2-b^2)c$。

程序如下：

```
10 HOME
20 LET A=6
30 LET B=5
40 LET C=100
50 LET V=(3.14/4)*(A∧2-B∧2)*C
60 PRINT "V="; V
70 END
RUN ↵
```

这时屏幕显示如下

V＝863.5

需要说明的是，50号语句是将计算出圆管体积赋于$V$；其公式中的$\pi$用3.14代替，则是因为学习机并不知道$\pi$的值；60号语句是显示$V$值；70号语句表示程序结束。

## 49.让学习机打印一张课程表

——介绍INPUT语句、PRINT语句和TAB函数

### (1) 键盘输入语句 INPUT

前面我们介绍了直接赋值语句 LET，并让学习机用它做了一道计算圆管体积的数学题。假如我们想要改变一组参数去求圆管的体积，就要修改程序，将三个 LET 语句中的语句体一一改动，比较麻烦。那么，有没有更简便的方法呢？利用键盘输入语句 INPUT 直接从键盘接收数据就是简便方法中的一种。

让我们用 INPUT 语句将上面例题改写一下：

```
10 HOME
20 INPUT  A
30 INPUT  B
40 INPUT  C
50 LET  V＝(3.14/4)＊(A∧2－B∧2)＊C
60 PRINT “V＝”；V
70 END
RUN ↵
```

屏幕上出现一个“？”

这是程序运行到20号语句INPUT A，学习机就在等待人从键盘上键入一个数赋于A。当我们键入一个具体的数值并回车后，学习机便自动地将此数赋于A，并继续往下执行，这时屏幕上又出现一个问号，是在等待从键盘上送入一个数值赋于B。

这样，要想随时更换一组参数，只要重新运行一下该程序，当屏幕出现问号，键入新的数值即可。

为了在操作时能够分清遇到一个问号是要给哪一个变量赋值，以免出错，INPUT 语句可以带有提示信息。

如20语句加上提示信息为：

20 INPUT “A=”；A

运行时屏幕出现“A=？”，使我们很清楚该为哪个变量赋值。

一个 INPUT 语句可以给几个变量赋值，变量与变量之间要用“，”分割。在用键盘输入数值时，数值与数值之间也要用“，”分割。

如上面程序20、30、40号语句可以合并为一条语句：

20 INPUT A，B，C

这样顺序键入三个数值分别赋于 A，B，C，注意次序不能搞混。

另外，INPUT 语句在从键盘向变量赋值时只能键入具体数值，不能键入表达式，从这一点来说，INPUT 语句并不能完全代替 LET 语句。

INPUT 语句的格式：

| 行号 INPUT　变量，变量…… |
| --- |

或

| 行号 INPUT“提示信息”；变量，变量…… |
| --- |

（2）输出语句 PRINT

前面我们已经用到PRINT 语句，它的作用也很明显，输出运算结果，同时也可以用于键盘计算。

所谓键盘计算，就是可以将学习机当作计算器来使用，其方法如下：

键入

PRINT 3+5 ↵

屏幕上立即出现 8

再键入

PRINT (3+2+5)/2 ↵

将显示

5

这种方式称为立即执行方式，它没有行号，回车后就立即执行，用 LIST ↵来显示时看不到任何东西，因为它不是程序行，并没有记入内存。

PRINT 语句有多种功能，请看下面程序：

10 x=1000

```
20 PRINT x
30 PRINT
40 PRINT —6
50 PRINT "ABCD"
60 PRINT 30+x
70 END
RUN ↲
```

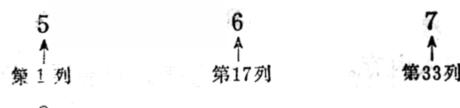
屏幕显示：

```
1000      显示变量的值
          显示一空行
—6        显示一常数值
ABCD      将字符串常量原样显示
1030      计算并显示表达式的结果（x的值为1000与30
          相加）
```

以上是PRINT语句的一般格式，它还能将运算结果以多种格式输出。

①标准格式：看下面程序。

```
10 PRINT 5，6，7，8
20 END
RUN ↲
```




上面的输出格式就为标准格式，程序中待显示项之间用

“，”分割，显示时，将屏幕40列分为三段，第一段从第1列至第16列　第二段从第17列至32列，第三段从第33列至40列，若待显示项为多于3个，转下一行第一段首位，上面10号语句的输出正是按照此规定。

②紧凑格式：看下面程序。

```
10 PRINT 19; 87
20 PRINT "AA"; "BB"; "CC"
30 END
RUN ↲
1987
AABBCC
```

以上显示就为紧凑格式，语句中待显示项之间用“；”分割，输出时项与项之间没有间隔，紧挨在一起。

C.不换行格式：看下面程序。

```
10 PRINT 1, 3,
20 PRINT 5
30 PRINT 2; 4;
40 PRINT 6
50 END
RUN ↲
```

屏幕显示

```
 1          3          5
246
```

由于10号语句末尾有“，”，所以20号语句显示项不换

行，显示在第三段33列上，又由于20号语句末尾没有任何符号，所以30号语句显示时换行；同样，30号语句末尾有“；”，所以后面不换行按紧凑格式显示。

(3) 输出格式函数——TAB函数

尽管PRINT语句可以将结果以多种格式输出，但假如我们想将字符或数字显示在预定位置，它就难以办到，这只有依靠TAB函数来实现。如：

```
10 HOME
20 PRINT TAB(2)；“星期一”；TAB(10)；“语文”
30 END
RUN ↵
```

屏幕显示

```
星期一      语文
第 第       第
二 三       十
列 列       列
```

由上例可以看出，TAB函数的使用格式为：

PRINT TAB（表达式1）；输出项1；TAB（表达式2）；输出项2……

其中，表达式的值规定了输出项的起始位置（其值不能是负值）。一个PRINT语句可用几个TAB函数，但中间要以“；”分割，并且其值只能从小到大排列，否则这个函数无效，只作紧凑格式处理。若其值为小数，则计算机自动取整，如2.5作2处理。

(4) 学习机开始打课程表

现在我们可以共同来完成打印课程表的工作。

课　程　表

| | 一 | 二 | 三 | 四 | 五 | 六 |
|---|---|---|---|---|---|---|
| 一 | 语文 | 数学 | 语文 | 数学 | 语文 | 数学 |
| 二 | 数学 | 语文 | 数学 | 语文 | 数学 | 语文 |
| 三 | 音乐 | 美术 | 体育 | 地理 | 自然 | 品德 |
| 四 | 地理 | 音乐 | 自然 | 美术 | 体育 | 班会 |
| 五 | 自习 | 活动 | 自习 | 活动 | 自习 | 活动 |

程序如下：

```
10 PR#1
20 PRINT TAB(15); "课 程 表"
30 PRINT  "…………………………………"
40 PRINT TAB(5);  "一"; TAB(11); "二";
   TAB(17); "三"; TAB(23);  "四"; TAB(29);
   "五"; TAB(35);  "六"
50 PRINT TAB(2);  "一", TAB(5);  "语文";
   TAB(11);  "数学"; TAB(17);  "语文"; TAB
  (23);  "数学";TAB(29); "语文"; TAB (35);
   "数学"
…………
```

…………

将这个程序接着做下去，就可以打印出一张课程表来。

其中第10号语句是连接打印机，这样就可以将上面程序输出结果送到打印机去打印了。

### 50.计算数列前几项立方的和

——介绍读数语句 READ，置数语句 DATA，恢复数据区语句 RESTORE，无条件转移语句 GOTO

（1）读数语句READ和置数语句DATA

前面已经介绍了两种赋值语句：直接赋值语句 LET 和键盘输入语句 INPUT。LET 语句一次只能为一个变量赋值，INPUT 语句虽然能为多个变量赋值，但这就要占用大量的时间。为了解决这个问题，BASIC 为我们提供了用于成批变量赋值的语句——READ和DATA语句。

我们先看下面一段程序，从这个例子里可以体会到READ和DATA语句的功能。

〔例〕 计算下面数列的和：2，5，46，65，87，103

程序如下：

```
10 HOME
20 READ A, B, C, D, E, F
30 LET S=A+B+C+D+E+F
40 PRINT S
50 DTAT 2, 5, 46, 65, 87, 103
60 END
```

由上面的程序我们可以看出，READ 和 DATA语句的格式：

| 〈行号〉 READ 变量1，变量2…变量$n$ |
|---|
| 〈行号〉 DATA 数值1，数值2…数值$n$ |

READ 语句中，变量与变量之间要用“，”分割，DATA语句中，数值与数值之间也要用“，”分割，注意最后不能有标点符号。

BASIC语言专门为 DATA 语句开辟了一个存贮数据的数据区，当DATA语句输入后，该程序运行结束，按回车键后即按照输入的先后次序排列，存贮在数据区里，所以与写程序的行号无关，可以放在程序的任何位置。READ 语句是执行语句，在程序运行过程中遇到 READ 语句，就按照变量的先后次序，从数据区读对应的数值。我们可以认为数据区设有一个指针，开始时指针指向第一个数值，读过一个数值之后指向下一个数值。前一个 DATA 语句行读完之后读下一个 DATA 语句行的数值。所以，DATA语句可以分作几个程序行来写。但是在 DATA 语句中的数值只能是常数，不能是分数和表达式等。同时，DATA 语句中的数值类型必须和READ语句中的变量类型一致。若 READ语句中的变量为字符型，如A$，那么在DATA语句中的数值一定也要是字符型，因此数值项要用“ ”括起来，如“C”。

另外，DATA语句中的数值个数不得少于READ语句中变量的个数，否则在程序执行过程中将产生数据不足的错

误。

(2) 恢复数据区语句(RESTORE)

我们看下面一段程序：

```
10 READ A, B, C
20 DATA 4, 5, 6
30 READ X, Y, Z
40 DATA 4, 5, 6
50 PRINT "A="; A, "B="; B, "C="; C
60 PRINT "X="; X, "Y="; Y, "Z="; Z
70 END
RUN ↵
A=4     B=5     C=6
X=4     Y=5     Z=6
```

我们看，A，B，C 和X，Y，Z所读的数据相同，那么我们能不能让它们共用一个数据区，在读完 A，B，C 的数据之后让数据区的指针恢复在数据区的前面呢？答案是：可以的。这就要用到我们下面要介绍的恢复数据区语句 RESTORE。

它的格式是：

| 行号　RESTORE |
|---|

我们用 RESTORE 语句将上面程序进行改写：

```
10 READ A, B, C
```

```
20 RESTORE
30 READ X, Y, Z
40 PRINT "A="; A, "B="; B, "C="; C
50 PRINT "X="; X, "Y="; Y, "Z="; Z
60 DATA 4, 5, 6
70 END
RUN ↵
A=4    B=5    C=6
X=4    Y=5    Z=6
```

由输出结果可以看出 RESTORE 语句起了作用。它将数据区的指针恢复在前面了。使用了 RESTORE 语句，可以减少数据的书写次数，同时也节约了内存空间。

有时候我们并不一定希望将数据区指针恢复到指向第一个数据，而只想恢复某几个数据，则可采取以下方法：

如果我们想把 4，5，6，7，8 五个数据前三个赋给变量 A，B，C，后三个赋给X，Y，Z

写出下列程序：

```
10 READ A, B, C
20 RESTORE
30 READ I, J, X, Y, Z
40 PRINT "A="; A, "B="; B, "C="; C
50 PRINT "X="; X, "Y="; Y, "Z="; Z
60 DATA 4, 5, 6, 7, 8
70 END
```

```
RUN ↵
A=4     B=5     C=6
X=6     Y=7     Z=8
```

结果符合题目的要求。那么30号语句中，I，J变量究竟起到什么作用呢？10号语句执行完后数据区指针指向 7，20号语句执行后，指针恢复指向第一个数据，根据题目要求，现在要将指针指向第三个数据6，指针需要向后移两个数字位。为达此目的，增加了两个变量I，J，作用只是“假读”，实际起移动指针的作用。

（3）无条件转向语句GOTO

计算机的重要作用就是代替人去做重复性工作。例如计算不同参数圆管的体积，需要将程序运行多次，人工干预的话，操作次数就会十分频繁。能不能将程序设计成能够自动地回过头去再次执行呢？GOTO语句可以办到。

现在我们将原来的程序加上 GOTO 语句，看它是如何执行的。

```
10 HOME
20 INPUT A, B, C
30 LET V=(3.14/4)*(A∧2-B∧2)*C
40 PRINT "V="; V
50 GOTO 20
60 END
RUN ↵
? 8, 6, 50 ↵
```

V=1099

?　4，3，80 ↵

V=439.6

?

……

从上面情况我们看出，只要键入一次 RUN，程序就会不断地等待你输入一组参数，并计算出相应的体积。它的执行顺序为10→20→30→40→50→20→30→40→50→20……如此循环下去，虽然60号语句END，表示程序在这里要结束，但是永远也执行不到这一行。那么怎样才能阻止它一直回转下去呢？目前的办法是只要同时按下 CTRL 、 C 键即可。

GOTO语句的格式：

| 行号 GOTO 行号 |
|---|

当计算机执行到 GOTO 语句时，不再顺序执行下去，而是跳到 GOTO 语句后面所指定的行号并继续顺序执行下去。

(4) 计算数列前N项立方和

有数列：3，7，16，23，56，69，97，108，214，325，……

我们打算求出前N（N=1，2，3，…）项立方之和，并且在读入一个数据后，显示出累加的立方和。注意：我们

先要将准备放累加值的单元S和放项数的单元N清零。程序如下：

```
10 S=0
20 N=0
30 READ A
40 S=S+A∧3
50 N=N+1
60 PRINT "数列的前"；N "项立方和="；S
70 GOTO 30
80 DATA 3，7，16，23，56
90 DATA 69，97，108，214，325
100 END
RUN ↲
数列的前 1 项立方和=27
数列的前 2 项立方和=370
      ⋮
      ⋮
数列的前10项立方和=
```

此时若键入 CTRL 、 C ，则屏幕显示：

BREAK IN 30

意思是程序在30号语句被中止。

我们将N单元作为计数单元，每循环一次增加1，可以为我们有效地计录求和的项数。S单元存放前N项立方和，每循环一次，将此项立方和与前N−1项立方和进行累加。

请大家考虑一下：若第10、20号语句分别给S、N赋一定值，后面将会出现什么结果，若70号语句改为GOTO 10，又将出现什么结果呢？

## 51.初步的程序设计小结及例题

通过前面的学习，我们大概对程序设计有了一定的了解。学习机在我们面前可能也不会是那么神秘了。在程序设计的道路上我们前进了一大步，需要停下来，回顾一下，看看能用已学到的知识干些什么，这将会带来更大的提高。

（1）程序的构成与如何装入执行

我们已经接触过不少的程序了，从中可以知道程序是由若干条语句行组成，每条语句是由语句标号、语句定义符、语句体组成。在键入一条语句之后要以 RETURN 键结束。

程序装入执行的大体步骤如下：

①清除内存、清除屏幕（NEW HOME）；

②键入程序，每行要以回车符结束；

③列程序清单(LIST)、检查程序、修改程序。

④运行程序（RUN）。

（2）初步程序设计语句小结

我们将初步程序设计所涉及的语句进行小结（表4-2）。

初步程序设计中涉及到的函数只有一个——输出格式函数TAB函数。它的格式为TAB($x$)，其中$x$规定了输出项的起始位置。

表 4-2

| 语句类别 | 语句定义符 | 语句格式 | 例子 | 说明 |
|---|---|---|---|---|
| 赋值语句 | LET | 行号 LET 变量＝表达式 | LET A=3<br>LET A $=“ABC”<br>LET S=3*A/C | 将常量赋给变量A<br>将字符串赋给字符串变量A $将等号右边的表达式计算出来的值赋给S |
| | INPUT | 行号 INPUT 变量，变量……<br>行号 INPUT “提示信息”；变量，变量 | INPUT A,A $ ,B<br>INPUTA“请键入A”;A | 在程序执行时，显示问号后，依次键入各变量值，以逗号分割<br>屏幕上显示出“ ”里的提示信息，然后键入变量值 |
| 赋值语句 | READ<br>DATA | 行号 READ 变量 1，变量 2 …<br>行号 DATA 数值1,数值2… | READ A，B,C,D<br>READ A $<br>DATA 1，2，3，4<br>DATA “LI” | READ语句在执行中从DATA语句中顺序取出数据，赋给相应的变量<br>注意类型要一致 |
| 输出语句 | PRINT | 行号 PRINT 表达式 | PRINT 5<br>PRINT X+Y<br>PRINT “A=”<br>PRINT A，B<br>PRINT A；B<br>PRINT | 显示常数<br>显示变量或表达式的值<br>显示字符串<br>以标准格式显示——两项首字符距离16列<br>以紧凑格式显示——两项之间没有距离<br>将光标跳至下行首列 |
| 恢复数据区语句 | RESTORE | 行号 RESTORE | | 将DATA数据区指针移向第一个数据 |
| 无条件转向语句 | GOTO | 行号 GOTO | GOTO 20 | 将程序转向20号语句，然后顺序执行 |

(3) 三种赋值语句的比较

在初步的程序设计里，我们已将三种赋值语句全部做了介绍，即直接赋值语句(LET)、键盘输入语句（INPUT），读数/置数语句（READ/DATA），那么，在我们程序设计中，要向变量赋值，究竟用哪一种语句好呢？下面我们加以比较。

①直接赋值语句（LET）：LET 主要用于计算。每条语句只能为一个变量赋值，但它赋值的速度比较快。

②键盘输入语句（INPUT）：键盘输入语句是在程序运行过程中，通过键盘为变量赋值，它可以在一条语句里为多个变量赋值，但由于人工的干预，速度比较慢，主要用于为经常需要人工改变数值的变量赋值。

③读数/置数语句（READ/DATA）：READ/DATA 语句主要用于为大批量的数据赋值，且速度快。但后两种赋值语句均没有计算功能。

(4) 程序的结构

程序的结构大体上是按照程序执行的路线来划分的。在初步的程序设计里，我们接触到的程序，除了由无条件转移语句 GOTO 所形成的无条件循环结构外，都是按照语句标号由小到大的顺序执行。这是计算机固有的执行路线，我们称此类程序结构为顺序结构。

下面，我们运用已学过的知识在计算机上实践实践。

〔例 1〕 让计算机与小朋友对话。

这个程序可以由年轻的爸爸妈妈们编写，也可以按照我

们下面编好的程序输入计算机，然后让孩子们和计算机对一对话。

程序的要求是计算机首先问："喂，小朋友，你叫什么名字呀？"当小朋友回答了自己的姓名（如果不会写出自己的姓名，写汉语拼音也行），计算机接着问："你几岁了？"小朋友可以接着回答。当按照计算机提出的问题一一回答完之后，计算机可以将孩子的回答显示在屏幕上。程序如下：

10 HOME

20 PRINT "小朋友，你好！你叫什么名字呀？"

30 INPUT A$

40 PRINT "你今年几岁了？"

50 INPUT B$

60 PRINT "你愿意和学习机交朋友吗！"（回答愿意或不愿意）；

70 INPUT C$

80 PRINT A$

90 PRINT B$

100 PRINT C$；"和学习机交朋友。"

110 END

RUN ↵

小朋友，你好！你叫什么名字呀？"

？张玲（操作者用键盘回答）

你今年几岁了？

？八岁（操作者用键盘回答）

你愿意和学习机交朋友吗？

？愿意（操作者用键盘回答）

张玲

八岁

愿意和学习机交朋友

〔例 2〕 有数列

$1，\frac{1}{2}，\frac{1}{3}，\frac{1}{4}，\frac{1}{5}，\cdots\cdots$

依次显示前 1 项之和，前 2 项之和，……前 $n$ 项之和。实际上显示的值为：

$$S_1=\frac{1}{1}$$

$$S_2=\frac{1}{1}+\frac{1}{2}$$

$$S_3=\frac{1}{1}+\frac{1}{2}+\frac{1}{3}$$

……

从项数来看，每次递增 1，设此变量为 $N$，$N=N+1$，而且要设初值 $N=0$，则第一次值为 1，依次为2，3，4…

设累加和的单元为 $S$，$S=S+\frac{1}{N}$，同样设初值 $S=0$，

这样 $S$ 的值第一次为 $\frac{1}{N}$ $\left(即\frac{1}{1}\right)$ 第二次为 $\frac{1}{1}+\frac{1}{2}$……。

程序如下：

10 HOME

20 $N=0$

30 $S=0$

40 $N=N+1$

50 $S=S+\frac{1}{N}$

60 PRINT “前”；N；“项和＝”；S

70 GOTO 40

80 END

```
            *
         *  *  *
      *  *  *  *  *
   *  *  *  *  *  *  *
```

图 4-2

〔例 3〕 编一个显示图 4-2 的程序。

我们定第一个星为第10列，程序如下：

10 HOME

20 PRINT TAB(10)；“*”

30 PRINT TAB(9)；“* * *”

40 PRINT TAB(8)；“* * * * *”

50 PRINT TAB(7)；“* * * * * * *”

60 END

# 二、进一步的程序设计

经过学习初步的程序设计，我们了解了程序设计的基本方法，能用计算机解决一些简单的应用问题。从程序的结构来看，初步的程序设计是按照程序执行的固有路线，即按照语句标号由小到大的顺序执行，或由GOTO语句构成反复回转的执行路线。但计算机之所以称为电脑，就是人赋于它逻辑判断的功能，我们只有发挥了它这一功能，才使计算机有了思维能力——根据不同的条件、不同的情况而采取不同的处理方法。前面学到的基本语句不能解决此类复杂的问题，需要学习进一步的程序设计。学习此类程序设计的本身，将会对学习者的逻辑判断和思维能力有很大提高。

进一步程序设计包括分支程序设计和循环程序设计，是整个BASIC程序设计的重要内容。

## 52.帮你分辨出是平年还是闰年

——介绍分支程序设计

我们经常遇到如下问题：夏天买西瓜，西瓜的价格与你买的多少有关，如果你买100斤以上，每斤为 8 分，否则每斤为 1 角。

我们不难写出计算西瓜金额的公式：

设 $x$ 为买西瓜的斤数，$y$ 为应付的钱数：

$$y=\begin{cases}0.08x & (x \geq 100)\\ 0.10x & (x < 100)\end{cases}$$

从上面我们可以看出，$y$ 的值取决于 $x$ 的取值范围，也可以说取决于一系列的条件。条件成立，即条件为真，就按照相应的公式去计算西瓜的金额。

我们再来看一个例子：我们知道平年为365天，闰年为366天，那么如何来计算某年是否为闰年呢？当某一年的年号不能被 4 整除时它就是平年；能够被 4 整除但不能被100整除时，这一年也是闰年；能够被 4 和100整除，并同时也能被400整除的年也为闰年，否则都是平年。

这里存在着一系列的条件：设年号为 $x$，则

①$x$ 能否被 4 整除；

②$x$ 能否被100整除；

③$x$ 能否被400整除。

必须经过多方判断，才能确定此年为闰年还是平年。那么如何将此判断方法教给计算机，让它代替人去自动地分辨出平年和闰年呢？这也如同我们在前进的道路上出现了三岔路口，看下一步要走哪条路：如果条件成立，则走条件为真的路。

“如果”在英语里用 IF 表示，“则”用THEN表示，可以写成：IF条件为真　THEN　走条件为真的路。

这就是条件语句 IF……THEN，它的格式如下：

格式 1

| 行号 IF 关系表达式 THEN 行号 |
|---|

意思是：如果条件成立，程序转向 THEN后面的行号所指明的语句去执行。否则，仍顺序往下执行。

格式 2：

| 行号 IF 关系表达式 THEN 可执行语句 |
|---|

意思是：如果条件成立，执行 THEN 后面的可执行语句。否则，按顺序往下执行。

条件用关系表达式来确定。如$x<50$就是关系表达式，它是由关系运算符连接而成的。关系运算符有：＜（小于），＞（大于），＝（等于），＞＝（大于或等于），＜＝（小于或等于），＜＞（不等于）等六种。

下面就可以着手对前面提出的两个问题，编制程序了。

第一个问题：计算买西瓜应付钱数的程序为：

```
10 HOME
20 INPUT "请输入西瓜的斤数";x
30 IF x>=100  THEN y=0.08*x: GOTO 50
40 y=0.10*x
50 PRINT "应付钱数为"; y; "元"
60 END
RUN ↵
```

请输入西瓜的斤数？80 ↵

应付钱数为 8 元

这个程序30号语句为IF—THEN语句，如果条件成立，按y＝0.08x 计算，并且转向50号语句去执行。在 THEN后有两条语句，第一条 y＝0.08＊x 为赋值语句，赋值语句可以省略语句定义符LET。第二条语句为GOTO 50，在同一行里出现多于一条的语句称为多语句行，中间要用“：”分开。

那么，在30号语句中，GOTO 50 起什么作用，去掉它行吗？不妨自己试一试。

第二个问题：分辨某年是平年还是闰年。

在处理这个问题的过程中，要用到一个函数——取整函数INT，它的作用是取不大于此数的最大整数。

例如：INT(3.5)＝3

INT(－4.5)＝－5

利用这个函数我们可以判断某数是否能被另一个数整除。比如1986能否被 4 整除，那就要看1986/4等不等于INT(1986/4)，如果相等了，就是能被 4 整除。

现在把判别平年和闰年的条件列举出来，设年号为N，则有 N/4＝INT(N/4)； N/100＝INT(N/100)； N/400＝INT(N/400)为条件。

程序如下：

```
10 HOME
20 INPUT "请输入年号"；N
```

```
30 IF N/4<> INT (N/4) THEN PRINT"平年":
   GOTO 70
40 IF N/100<> INT (N/100) THEN PRINT "闰
   年": GOTO 70
50 IF N/400=INT (N/400) THEN PRINT "闰
   年": GOTO 70
60 PRINT "平年"
70 END
RUN ↵
? (请输入年号) 2020
闰年
RUN ↵
? (请输入年号) 1985
平年
```

因为这个程序条件比较多，再来仔细分析一下：30号语句表示N不能被4整除，条件成立，就是平年了。若条件不成立，即N能被4整除接着向下执行；40号语句表示N已经能被4整除了，但不能被100整除，就是闰年；50号语句，$N$能被4整除，也能被100整除同时也能被400整除，也为闰年；60号语句，除此之外，都为平年。

## 53.学习机会为我们排序

——介绍程序流程图

将许多数据按照一定的规律排列出来，这是件比较头疼

的事，数越多，排起来速度就越慢，能不能让学习机来干呢？

先从简单问题开始。将三个数按照由大到小的顺序排列出来。人来处理这个问题往往采取两两比较，若写出程序，必然要多处出现分支，条件语句使程序结构发生了变化，不再按程序固有的路线执行，这样，程序设计起来就复杂了，为了使思路清晰，我们要借助于程序设计的辅助工具——流程图（框图）把程序执行的路线表示出来。

流程图画法的具体规定为：

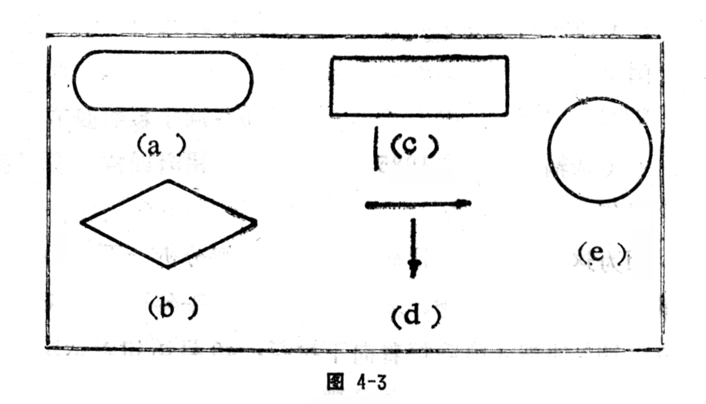



图 4-3

（1）开始，结束框：〔图4-3(a)〕表示程序的开始或结束。

（2）处理框：[图4-3(c)]：表示程序所要进行处理和运算的工作。

（3）判断框〔图4-3(b)〕：表示程序要进行判断的内容。它有一个入口，两个出口。一个出口为条件满足，用

“Y”（YES简写）表示；另一个出口为条件不满足，用“N”（NO的简写）表示。

(4) 流程线〔图4-3(d)〕：表示程序执行的路线。

(5) 连接点：〔图4-3(e)〕：中间标以数字，表示流向框外的某处，或由框外的某处进入该框图。

〔例〕 计算房租。若住房面积等于或小于40平方米，按每平方米0.10元计算，而住房面积大于40平方米的，超出部分，按每平方米0.50元计算。

设住房面积为$N$，房租为$S$，计算公式为：

$$S=\begin{cases}0.10N & N\leqslant 40\\ 0.10\times 40+0.5\times(N-40) & N>40\text{。}\end{cases}$$

框图如图 4-4。其中第一个判断框 N＝－1 是设立的一个结束标志，如果不再想计算了，这时，便从键盘上将 －1 赋给$N$，这时此条件成立，程序转向结束。

程序如下：

```
10 INPUT “请输入住房面积”；N
20 IF N=-1 THEN 70
30 IF N>40 THEN S=0.1*40+0.5*(N-40)：
   GOTO 50
40 S=0.10*N
50 PRINT “应付房租”；S；“元”
60 GOTO 10
70 END
```

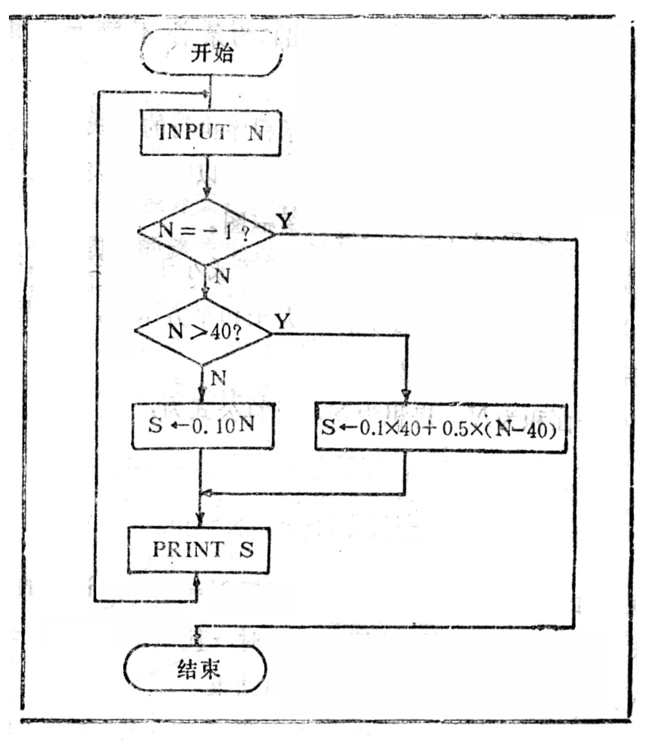



图 4-4

RUN ↵

请输入住房面积？30

应付房租 3 元

请输入住房面积？50

应付房租9元

请输入住房面积？ -1

这时，程序就执行完毕。

对于编制较为复杂的程序，一定要先画出框图，然后再着手写程序，那么如何将框图变为程序呢？

当把框图画好之后，对照着框图中每个框叙述的内容，看看要用什么样的语句来实现。关键是看看有几个分支，各分支之间的关系。注意找出程序主要的执行路线(即主干)，然后一个分支，一个分支地将程序写出，要特别注意转移的语句标号一定要对应。

下面我们将以计算机排序为例来实践这一过程。

我们知道，三个数的排列，可以有六种不同的顺序，我们采取两两互相比较的方法进行。框图如图4-5。

乍一看，这个框图好复杂呀！要编写程序，如何下手呢？

其实，当你画出了框图，程序基本上就出来了。让我们一起来完成它吧！

这个框图，我们将其分为六个支路（图 4-5 ）。在这些支路中，我们首先找出主干①，先将①的程序写出：

```
10 INPUT "A="; A
20 INPUT "B="; B
30 INPUT "C="; C
40 IF A>=B THEN
50 IF B>=C THEN
60 PRINT C, B, A
700 END
```

只剩下40、50号语句中，转向何处的入口语句标号没有定下，我们暂且空着。

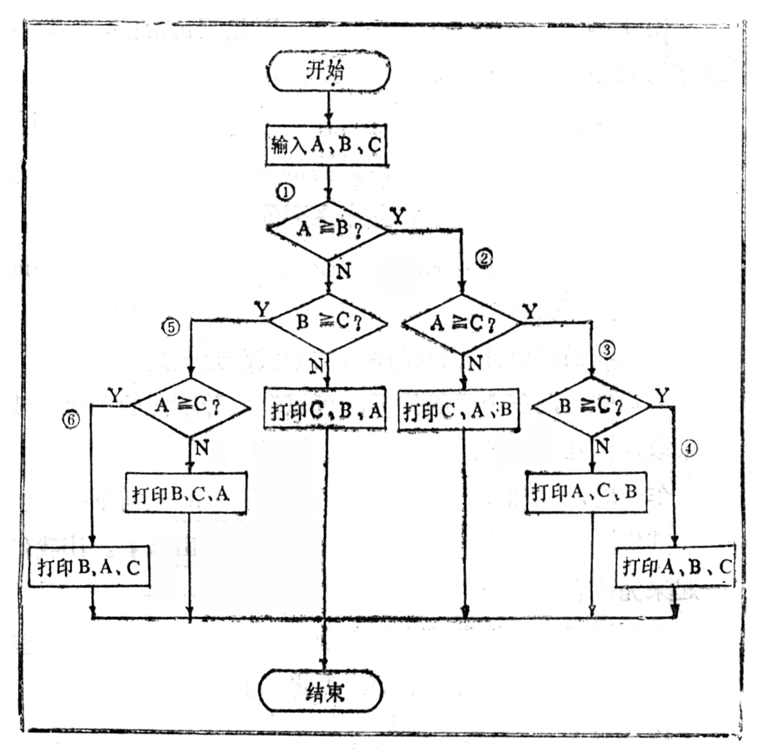



图 4-5

继续编写②支路的程序

```
80 IF A>=C THEN
90 PRINT C, A, B
100 GOTO 700
```

由于②支路入口地址明确了，所以40号语句的入口地址也就可以确定了。我们就可以将40号语句 THEN 后填入80了。注意，为什么100号语句要 GOTO到700呢，700号语句

体为 END，它是各支路的汇合点。

照此很容易就可以把各支路程序写出，然后将空着的语句标号填上，这个程序基本上就算完工了。

但是，这个程序的解题思路并不是最佳的。我们还可以用交换变量的方法来实现。这是一种常用的办法，具体是：首先确定在A这个单元放最大的数，方法是：A先与B比较，如果A大则不交换，A小则A、B交换。然后，A与C比较，A大则不交换，A小则A、C交换，确定A为最大。然后B与C比较，B大则不交换，B小则B、C交换；确定B为第二，那么C必然为最小。因此从大到小的顺序为A、B、C。画出框图（图4-6）。

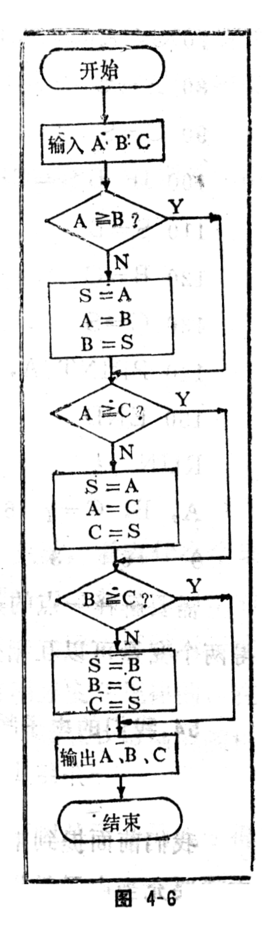


图 4-6

程序如下：

```
10 INPUT "A, B, C="; 
   A, B, C
20 IF A>=B THEN 60
30 S=A
40 A=B
50 B=S
60 IF A>=C THEN 100
```

```
70 S=A
80 A=C
90 C=S
100 IF B>=C THEN 140
110 S=B
120 B=C
130 C=S
140 PRINT A, B, C
150 END
RUN ↵
A, B, C=? 6, 9, 3
9    6    3
```

需要解释一点的是，程序中变量S是一个中间变量，使得两个变量可以互相交换。

## 54.我们的电子教师

——介绍 RND 函数

我们前面提到中华学习机可以为我们辅导功课。下面，它就要充当电子教师给大家出一些题目，检查、考试一番，看看知识掌握得怎么样了。考试的关键是出题目，题目要出得灵活，不能千篇一律。怎样才能做到这一点呢？不要着急，BASIC 里有一个有趣的函数——RND($x$)，它被称为随机数发生器，可以为你解除此烦恼。

由于 RND 函数可以随时给出变化莫测的数字而取名随

机函数。RND($x$)中的$x$值只是形式上的变量，它的取值范围因不同机型而有所不同。在中华学习机上，当$x$取正数时RND($x$)才真正产生随机数。

看下面的例子：当赋于$x$不同的10个值时，分别显示出它们的函数值：

```
 5 I=0
10 INPUT X
20 PRINT RND(X)
30 I=I+1
40 IF I<10 THEN GOTO 10
50 END
```

我们从键盘上分别键入正数、零、负数，看有什么结果出现。其结果是：当输入负数总是出现0～1之间某一固定的数。当输入零时，和上次结果相同。只有当输入正数时，才会不断出现一个与输入数无关的、及有明显变化规律的、在0～1之间均匀分布的数。那么怎样才能产生其它范围的随机数呢？这可以通过一定的运算来实现。

例如，产生0～99之间的随机整数，可以从下面的式子得到。

```
A=INT(100*RND(1))
```

若要产生0～99之间的带有两位小数的随机数，则

```
A=INT(10000*RND(1))/100
```

总之，掌握了以上这些，我们就可以编写程序，让学习机充当电子教师了。

我们准备来加强一下孩子们的口算能力，让他们任意挑选“+”、“-”、“×”、“÷”四种运算中的一种，然后由电脑出 100 之内的此种运算口算题10道，让孩子迅速给出答案，若全算对了，学习机给出“很好”；算对八道以上，学习机给出“好”；算对六道以上，学习机给出通过；算对六道以下，学习机会问你“愿不愿意再继续练习”，若回答“愿意”学习机会重新再给你出10道题。框图如图 4-7。

程序如下：

```
10 N=0：S=0
20 INPUT “准备好了吗？你要进行+、-、*、/ 哪
   一种运算的练习，请回答运算符”；T$
30 IF N>=10 THEN 150
40 A=INT(RND(1)*100)
50 B=INT(RND(1)*100)
60 IF T$= “+” THEN D=A+B：GOTO 100
70 IF T$= “-” THEN D=A-B：GOTO 100
80 IF T$= “*” THEN D=A*B：GOTO 100
90 IF $T= “/” THEN IF B=0 THEN GOTO
   50
95 D :INT(A/B+0.5)
100 PRINT A；T$；B；“=”；
110 INPUT C
120 IF C=D THEN S=S+1；PRINT“对了！”
    :GOTO 30
```

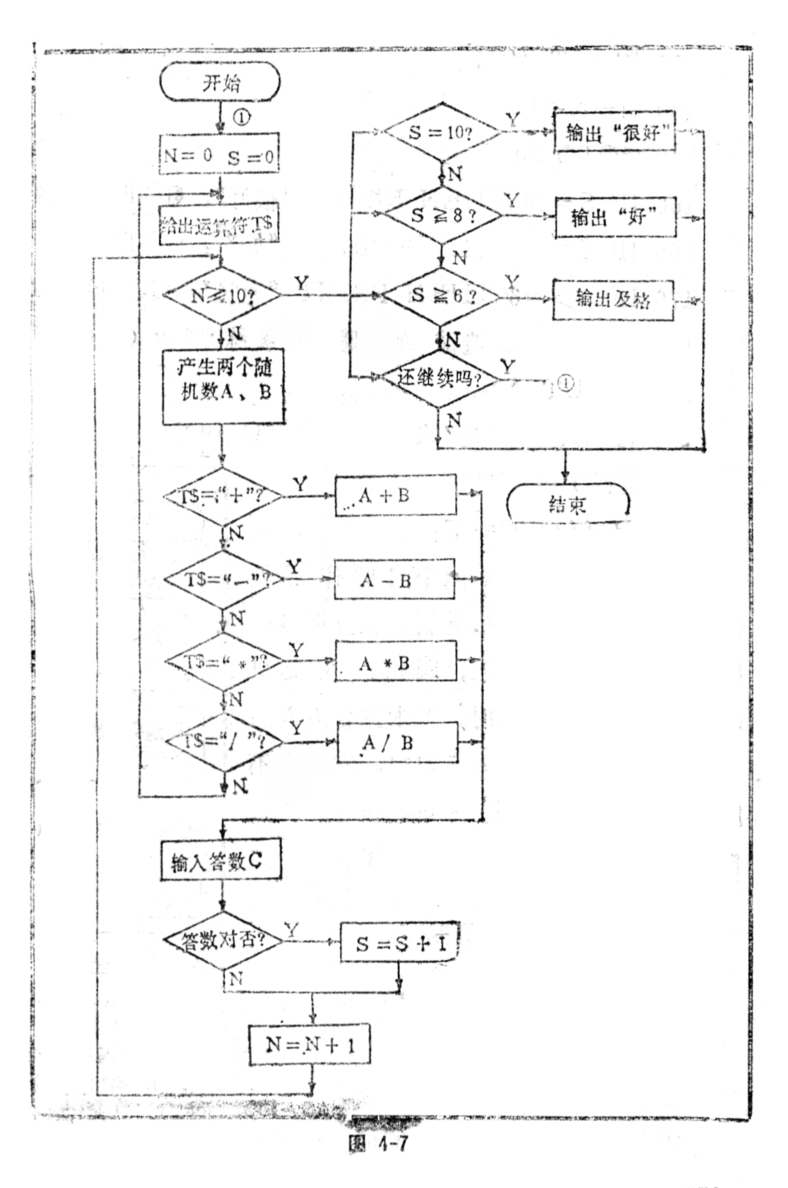

开始
①
N=0 S=0
给出运算符T$
N≥10?
Y
N
产生两个随机数A、B
T$="+"?
A+B
T$="-"?
A-B
T$="*"?
A*B
T$="/"?
A/B
输入答数C
答数对否?
S=S+1
N=N+1
S=10?
输出"很好"
S≥8?
输出"好"
S≥6?
输出及格
还继续吗?
①
结束

图 4-7

```
125 PRINT "错了！"
130 N=N+1
140 GOTO 30
150 IF S=10 THEN PRINT "很好"：GOTO 210
160 IF S>=8 THEN PRINT "好"：GOTO 210
170 IF S>=6 THEN PRINT "通过"：GOTO 210
180 PRINT "很遗憾,你还愿意继续练习吗？(是/否)"
190 INPUT A$
200 IF A$="是" THEN GOTO 10
210 END
```

将90号语句说明一下：

90号和95号语句对两个问题作了处理。一个是分母为零时，重新产生分母B；第二是将答案四舍五入，因此，在上机时，作除法不能整除时，要求四舍五入。

当我们将程序输入学习机，它就会不厌其烦地为我们出题、考试、改卷，俨然像一位非常负责任的"教师"了。

## 55.分支程序设计例题

［例 1］ 计算 Y=X！

框图如图4-8。

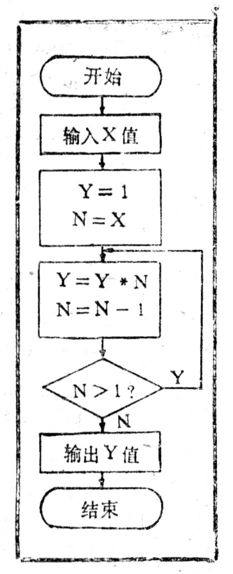



图 4-8

注：Y是存放连乘积的，因此要赋初值为1；N是计数单元。

程序如下：

```
10 INPUT "请输入x的值"；x
```

```
20 Y=1
30 N=X
40 Y=Y*N
50 N=N-1
60 IF N>1 THEN 40
70 PRINT"Y=";X; "!=";Y
80 END
RUN ↵
请输入x值? 3 ↵
Y=3!=6
```

[例 2] 输入若干个数，分别统计正、负数的个数及合计值。

在向学习机输入数时，我们设立一个终止标志，为-1，当学习机接收-1后，读数结束。

我们先将框图画出来（图4-9）。

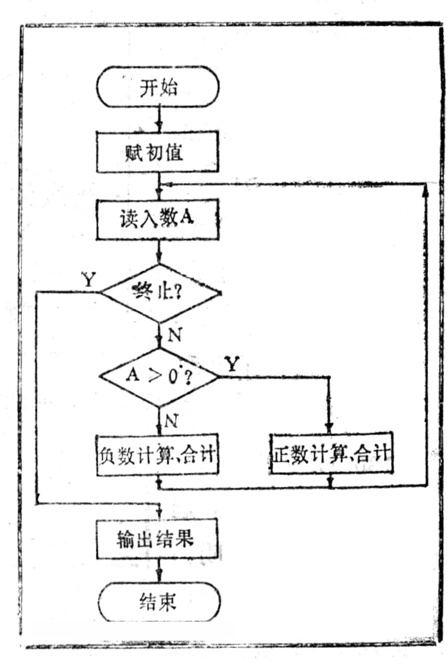



图 4-9

程序如下：

```
10 N1=0:S1=0:
   N2=0:S2=0:
20 INPUT A
30 IF A=-1
```

```
    THEN 110
40 IF A<0 THEN 80
50 N1=N1+1
60 S1=S1+A
70 GOTO 20
80 N2=N2+1
90 S2=S2+A
100 GOTO 20
110 PRINT "共有"; N1; "个正数", "它们的和是"; S1
120 PRINT "共有"; N2, "个负数", "它们的和是"; S2
130 END
```

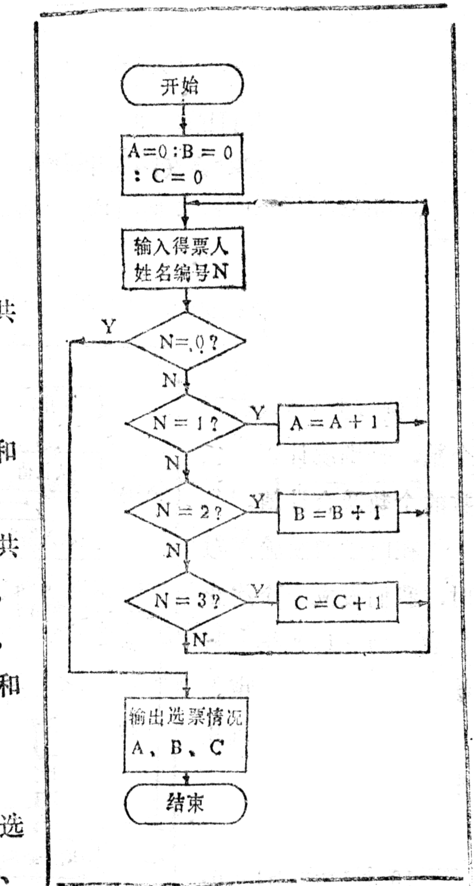



图 4-10

[例 3] 统计选票。现有王平、李明、张强三候选人，选其中一名为优秀队员，将选票情况显示出来。

框图如下：

程序如下：

```
10 A=0: B=0: C=0
20 INPUT "请输入得票人姓名(王平: 1, 李明:
   2, 张强: 3, 完毕: 0)"; N
30 IF N=0 THEN 70
40 IF N=1 THEN A=A+1: GOTO 20
50 IF N=2 THEN B=B+1: GOTO 20
60 IF N=3 THEN C=C+1: GOTO 20
70 PRINT "王平得票数为"; A
80 PRINT "李明得票数为"; B
90 PRINT "张强得票数为"; C
100 END
```

## 56.用学习机来验证实验数据

——介绍单重循环FOR—NEXT语句

循环程序的设计我们在前面已经有所接触，例如，求1～10的累加值。

在初步的程序设计里，可以写出如下程序：

```
10 S=0: I=1
20 S=S+I
30 PRINT S
40 I=I+1
50 GOTO 20
60 END
```

这个程序由GOTO语句形成往返重复，只有键入CTRL—C，才能够停止执行，而且循环的次数预先不能确定，因此，题目的要求也没有完全达到。

在分支程序设计里，又可以写出如下程序：

```
10 S=0：I=1
20 S=S+I
30 I=I+1
40 IF I>10 THEN PRINT S：GOTO 60
50 GOTO 20
60 END
```

看这个程序，是由IF和GOTO组成了循环。I既作为计数单元，又作为变量出现。先给I赋初值为1，执行一遍累加计算，然后修正计数单元，然后判定计数单元是否超出界限——若没有超出界限，再次累加；若超出了界限，循环结束。

我们知道，计算机最突出的优点是能代替人作大量重复性的工作，几乎所有的程序都要涉及循环过程，因此，BASIC语言提供了专门的循环语句FOR—NEXT来完成控制循环次数的工作。

下面我们就用FOR—NEXT语句来将上例程序写出：

```
10 S=0
20 FOR I=1 TO 10 STEP 1
30 S=S+I
40 NEXT I
```

```
50 PRINT S
60 END
```

这个程序看起来简单明了，而且执行速度快。其中的20号语句FOR开始为循环控制变量I赋初值为1，TO后面将循环控制变量的终值定为10，STEP为步长（即循环控制变量的增值，这里为1）。接下去，30号语句计算累加值，这是循环的主题部分；每循环一次，就要执行一次，因而称之为循环体，它处于FOR与NEXT之间。在这个程序里，循环体只有一条语句（当然也可以是多条）。接下去NEXT I为循环结束语句，表明此次循环结束，取循环控制变量的下一个值，再与FOR语句指出的循环变量终值进行比较。若没有超过终值，返回执行循环体；超出终值，退出循环，执行NEXT后面的语句。既然FOR语句和NEXT语句共同来完成控制循环次数的任务，因此，它们必须成对出现，否则计算机将给出出错信息（参看附录C）。

FOR—NEXT语句的格式如下：

```
FOR 循环变量=e1 TO e2 STEP e3
循环体
NEXT 循环变量
```

其中：$e_1$：初值；$e_2$：终值；$e_3$：步长。

执行过程为：先给循环变量赋初值$e_1$、终值$e_2$、步长$e_3$；执行循环体后，循环变量加上步长与终值比较，若大于终值

则循环结束，若小于终值，继续执行循环体。

循环语句在使用中要注意以下几点：

①循环变量的初值、终值、步长可以是常数、变量和表达式。当步长为1时，可以省略不写。

［例］　输出从0°开始到180°每隔10°的正弦值。

程序如下：

```
10 FOR I=0 TO 3.14 STEP 3.14/18
20 PRINT SIN(I),
30 NEXT I
40 END
```

其中：SIN(I)为正弦函数，角度在运算中采用弧度制。

以上举的例子步长都为正数，下面举一个步长为负数的例子。

［例］　要将1到50之间的奇数，由大到小按顺序输出，其程序如下：

```
10 FOR I=50 TO 1 STEP -2
20 PRINT I-1
30 NEXT I
40 END
```

第1次执行循环体，输出49，第二次输出47，第25次循环时I=2，执行循环体后，30号语句NEXT I，加上步长−2，当I=0时，超出终值，退出循环。

②循环变量初值、终值和步长选择要合适，否则会影响循环的执行。如

```
10 READ N
20 FOR I=3 TO -3 STEP N
30 PRINT I; " ";
40 NEXT I
50 GOTO 10
60 DATA -2, 0, 2
70 END
RUN ↙
3  1  -1  -3
3 3 3 3 3 3 ……
```

取第一个数据-2时，步长为-2，程序正常执行4次。取第二个数0时，循环无休止，因此不可取。当取第三个数+2时，步长为2，I=3时执行了一次循环体的运行，而当I再增加步长2使I=5时，离终止-3越来越远，循环无法执行。

③在执行循环语句的过程中，只允许由循环体内转向循环体外，而不允许由循环体外转向循环体内。

现在，让我们共同来完成一个题目：用计算机来验证实验结果。

题目内容是这样的：以初速度为120米/秒竖直上抛一物体，列出从最大高度开始，每下降1/10高度所用的时间。

A.分析：竖直上抛物体高度为$h=v_0t-\frac{1}{2}gt^2$；下落高

度为 $h=\frac{1}{2}gt^2$， $t=\sqrt{2h/g}$。

B.程序：

```
10 V0=120
20 G=9.8
30 T=V0/G
40 H=V0*T-1/2*G*T*T
45 PRINT "V0=" ;V0, "h=" ; H
50 FOR I=0 TO H STEP H/10
60 T=SQR(2*I/G)
70 PRINT"H="; I, "T=" ; T
80 NEXT I
90 END
```

## 57.判素数

### ——介绍多重循环

当我们学习过循环语句之后，了解到循环语句用于重复地做相同或类同的事情。而如果我们在循环内部还要做重复的相同或类同的事情时怎么办呢？这就要用到双重循环，甚至是多重循环。

［例］ 编写一个程序，能够打印出《九九表》（图4-11）。

从图4-11中可以看出：九九表具有10行10列。每一个循环输出一行，因此，要设计成在循环里面嵌套循环。程序

|  | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 |
|---|---|---|---|---|---|---|---|---|---|
| 1 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 |
| 2 | 2 | 4 | 6 | 8 | 10 | 12 | 14 | 16 | 18 |
| 3 | 3 | 6 | 9 | 12 | 15 | 18 | 21 | 24 | 27 |
| 4 | 4 | 8 | 12 | 16 | 20 | 24 | 28 | 32 | 36 |
| ⋮ | ⋮ | ⋮ | ⋮ | ⋮ | ⋮ | ⋮ | ⋮ | ⋮ | ⋮ |

图 4-11

为：

```
10 PR#1
20 FOR I=1 TO 9
30 PRINT TAB(I*5); I;
40 NEXT I
50 PRINT
60 FOR I=1 TO 9
70 PRINT I;
80 FOR J=1 TO 9
90 PRINT TAB(5*J); I*J;
100 NEXT J
105 PRINT
110 NEXT I
120 PR#0
130 END
```

若运行此程序，便在纸上打印出九九表。

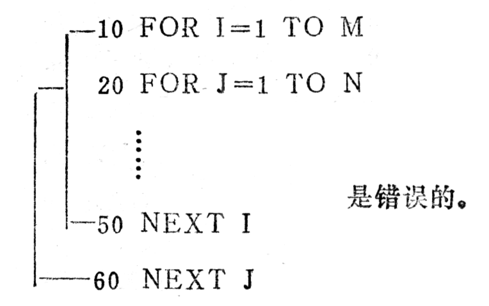
这个程序10号语句告诉学习机以后所有显示语句都打印

在纸上。120 号语句通知学习机返回屏幕显示。20～40号语句构成循环，打印出第一行的乘数，其间隔为 4 列；60～110号语句构成双重循环，它的功能是打印出被乘数和积。每执行一次外循环，就要将内循环执行一遍。即打印出一行，当外循环执行完，九行便全部打印出。

BASIC语言对多重循环作了以下规定：

①多重循环的层数最多不得超过八层。

②内外层循环不能交叉，如



```
10 FOR I=1 TO M
20 FOR J=1 TO N
   ⋮
50 NEXT I
60 NEXT J
```

是错误的。

下面我们共同来完成多重循环的例题。

[例 1]　统计一个班级每个学生五门功课的平均成绩（以六个学生为例）。如表4-3。

表 4-3

| | 语文 | 数学 | 外语 | 物理 | 化学 |
|---|---|---|---|---|---|
| 1 | 86 | 72 | 93 | 80 | 87 |
| 2 | 92 | 90 | 86 | 89 | 94 |
| 3 | 72 | 65 | 76 | 82 | 70 |
| 4 | 80 | 78 | 85 | 89 | 90 |
| 5 | 90 | 94 | 88 | 92 | 93 |
| 6 | 65 | 67 | 56 | 45 | 32 |

我们将计算每个学生五门功课成绩在内循环里运行处理。程序如下：

```
10 PRINT "学号", "平均成绩"
20 FOR I=1 TO 6
30 S=0
40 FOR J=1 TO 5
50 READ A
60 S=S+A
70 NEXT J
80 PRINT I, INT(S/5*10+0.5)/10
90 NEXT I
100 DATA 86, 72, 93, 80, 87
110 DATA 92, 90, 86, 89, 94
120 DATA 72, 65, 76, 82, 70
130 DATA 80, 78, 85, 89, 91
140 DATA 90, 94, 88, 92, 93
150 DATA 65, 67, 56, 45, 32
160 END
```

其中第80号语言在计算平均数时，四舍五入保留一位小数。

[例 2] 判素数。

我们准备将1～100之间自然数中的全部素数挑出来。所谓素数，就是除了1和此数本身之外，不能被其它整数整除的自然数。

我们知道除 2 以外，偶数不会是素数，那么只要查看奇数就行了，若此奇数 $N$ 能够被 3、4……直到INT$\sqrt{N}$整除，就应被排除在素数之外了。那么为什么只检查到 INT$\sqrt{N}$呢？那是因为假设$N$为合数，则有 N$=ab$，我们假定$a\leqslant b$，则有$a\leqslant\sqrt{N}$。

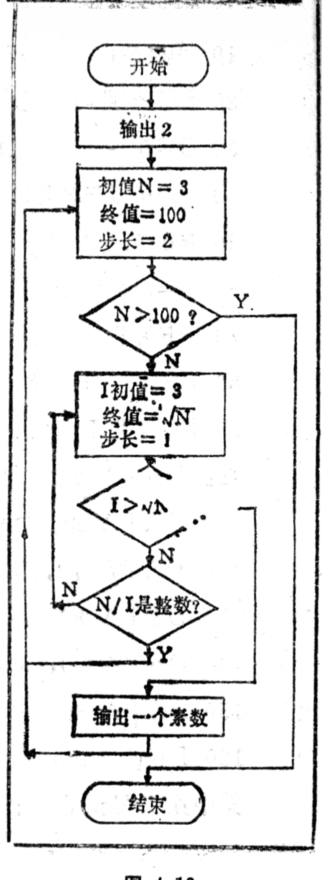



图 4-12

先将框图画出来(图4-12)。

程序如下：

```
10 PRINT 2,
20 FOR N=3 TO 99
   STEP 2
30 FOR I=3 TO INT
   (SQR(N))
40 IF N/I=INT(N/I)
   THEN 70
50 NEXT I
60 PRINT N,
70 NEXT N
80 END
```

## 58.进一步的程序设计小结及例题

进一步的程序设计是我们学习程序设计中最关键的一

环，它交给了我们程序设计有力的工具。

框图法是利用框图帮助我们理清思路，明白过程的流向，从而为设计复杂的程序奠定基础。

进一步的程序设计中，出现了分支程序结构和循环程序结构，它们都是最主要的程序结构，使计算机有了判断能力；同时，我们涉及到一些语句和一些函数，如表 4-4 、表 4-5。

表 4-4

| 语句类别 | 语句定义符 | 语句格式 | 例子 | 说明 |
|---|---|---|---|---|
| 条件语句 | IF……THEN | IF 关系表达式 THEN 行号<br>IF 关系表达式 THEN 可执行语句 | IF A>5 THEN 30<br>IF A$ = "Y" THEN PRINT A | 当关系表达式成立转向 THEN 后语句标号指向的语句<br>当关系表达式成立，执行 THEN 后可执行语句 |
| 循环语句 | FOR……NEXT | FOR 循环变量 $=e_1$<br>TO $e_2$ STEP$e_3$<br>循环体<br>NEXT 循环变量 | FOR I=1 TO10 STEP 2<br>PRINT I<br>NEXT I | $e_1$为循环变量初值<br>$e_2$为循环变量終值<br>$e_3$为步长 |

表 4-5

| 函　数 | 含　　义 | 例　　子 | 说　　明 |
| --- | --- | --- | --- |
| INT(X) | 取整函数，取不大于 $x$ 的最大整数 | INT(5.8)=5<br>INT(−5.8)=−6 | 1. 判断A能否被B整除<br>INT(A/B)=A/B<br>2. 用于四舍五入运算<br>INT(A*10+0.5)/10<br>小数点后第二位进行四舍五入 |
| RND(X) | 随机函数产生一个随机数 | RND(1) 产生一个(0,1)之间的随机数 | 产生一个0～99的随机整数<br>A=INT(RND(1)*100) |
| SQR(X) | 开平方 | SQR(9)=2 | 自变量 $x$ 的值必须大于或等于 0。 |

[例 1] （分牛的传说）相传古印度有一位老人，他临死前把三个儿子叫到床前说：“我的遗产只有十九头牛，你们分了吧！老大分总数的1/2，老二分总数的1/4，老三分总数的1/5，但切记不准将牛宰了”。说完便死了。

三个儿子左思右想，没有什么办法。有一天，一位老人牵着一头牛从他们家门前路过，看见三兄弟唉声叹气，问明原因，稍加思索说：“这样吧，将我这头牛借给你们，凑成二十头，老大分1/2应得十头，老二分1/4应得五头，老三分1/5应得 4 头，余下一头还归我。”三兄弟听罢非常高兴，问题得到圆满解决。

现在我们想想，似乎有些问题，老大应分19×1/2=9.5头，怎么分了十头？老二老三也不应该分那么多。现在，让计算机来给他们三兄弟重新分一下，看结果如何。

共有19头牛，第一次分：$19\times\left(\frac{1}{2}+\frac{1}{4}+\frac{1}{5}\right)=19\times\frac{19}{20}<19$（没有将牛分完），剩下$19-19\times\frac{19}{20}=19\left(1-\frac{19}{20}\right)=\frac{19}{20}$（头）。

第二次分：$\frac{19}{20}\times\left(\frac{1}{2}+\frac{1}{4}+\frac{1}{5}\right)=\frac{19}{20}\times\frac{19}{20}$（仍没分完），剩下$\frac{19}{20}-\frac{19}{20}\times\frac{19}{20}=\frac{19}{20}\times\left(1-\frac{19}{20}\right)=\frac{19}{20^2}$。……

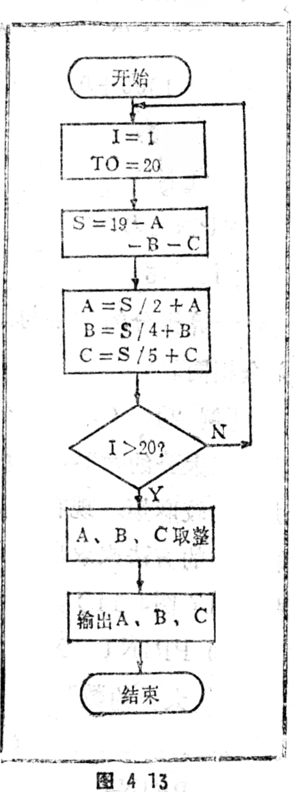



图 4 13

这样一次次地分下去，永无止境，我们假设分20次，设三个儿子分得数量为 A、B、C，为了得到整牛要进行四舍五入处理。先将框图画出来（图4-13）并设计程序：

```
10 FOR I=1 TO 20
```

```
20 S=19-A-B-C
30 A=S/2+A
40 B=S/4+B
50 C=S/5+C
60 NEXT I
70 PRINT INT((A*1000+0.005)/1000), INT((B
   *1000+0.005)/1000), INT((C*1000+0.005)/
   1000)
80 END
RUN ↲
10      5      4
```

计算机分的结果与老人分的结果完全相同。

［例 2］（求最大公约数）用手算求最大公约数时，一般采用短除法，比较麻烦。现在准备让学习机采取这样的方法来求出最大公约数。先看框图（图4-14），再设计程序：

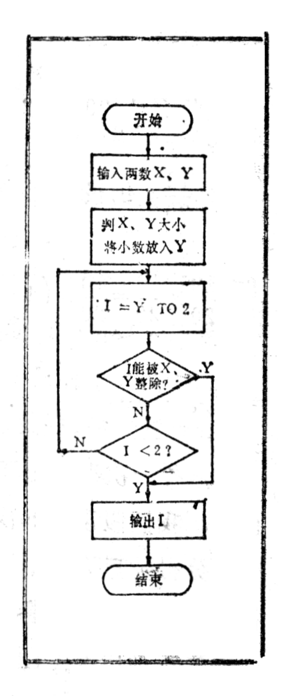


图 4-14

```
10 PRINT "X", "Y", "最
   大公约数"
20 INPUT X, Y
30 PRINT X, Y,
40 IF X>=Y THEN 60
50 C=X: X=Y: Y=C
```

```
60 FOR I=Y TO 2 STEP -1
70 IF (X/I=INT(X/I)) AND ((Y/I=INT(Y/I))
   THEN 90
80 NEXT I
90 PRINT I
100 END
```

[例 3] 打印出图4-15。

要打印出 I 行 J 列的图形，必然要用到双重循环（外循环控制行，内循环控制列）。外循环变量好确定，关键是内循环变量：确定每行列数，第一行为 1 列，第二行为 3 列，……第I行为 2×I－1 列。然后是花的位置要由 TAB 函数来确定。将第一个花打印在20列位置，第二行、第一个花为19列，因此为20－I列。

```
       *
      ***
     *****
    *******
   *********
  ***********
 *************
***************
```

图 4-15

程序如下：

```
10 FOR I=1 TO 8
20 PRINT TAB(20-I);
30 FOR J=1 TO 2*I-1
40 PRINT "*";
50 NEXT J
60 PRINT
70 NEXT I
```

```
80 END
```

## 三、高一级的程序设计

经过初级程序设计和进一步的程序设计之后，我们已经掌握了程序设计的基本方法。那么怎样才能使我们编制出的程序更有效、更简洁、更合理呢？那就是高一级程序设计所要解决的问题了。

### 59.评选最佳运动员

—介绍单下标变量和一维数组

我们以前学到的变量都是简单变量，它们之间没有什么内在的联系，它们各自有自己的名子。而学习机要处理的对象是成千上万的数据。例如我们要给1000个数赋值，面对着这样大批的数据，如何有效地发挥学习机的特点呢？这就要在这些数据上做些文章。我们把有内在连系的一批数进行编组，成为一组数，称之为数组。给它们取相同的名子，这就是数组名。但又要给它们进行编号，这就是下标。

例如将20名学生的数学成绩编为一组，命名为A，那么第一位学生的数学成绩就为A(1)，第二位学生的数学成绩为A(2)……，第20位学生的数学成绩为A(20)。

那么 A(1)、A(2)、……A(20)，分别称为下标变量。括号内的1、2、……20，分别称为下标，下标除了可以是常数以外，还可以是变量、表达式，如A(I)，B(I+J)等；而

且下标变量本身也可以作为下标出现，如 M(A(3))。但这些变量，表达式等预先要有确定的值，并且只能是正数和零，不能是负数，若带有小数，计算机会自动取整。

数组在参加运算之前要存放在内存中，那么究竟要给数组留多大的存贮空间呢？必须事先通知计算机，其方法是通过数组说明语句来实现。格式如下：

| 行号 DIM 数组名（数组元素下标的最大值或上界） |
|---|

例如，刚才我们举的表示20名学生数学成绩的数组 A，就可以这样来说明：

10 DIM A(19)

如果在同一程序中需要使用几个数组时，可以在数组说明语句中同时对这些数组进行说明，但之间要用逗号分开。如：

10 DIM A(10), B(25), C$(15)

如果下标值不超过10时，不用事先说明，可以直接引用。

那么究竟引用了数组和下标变量之后，能为我们编程带来什么方便呢？我们不妨来做个比较吧！

前面我们在初级程序设计里，曾对三个数进行从大到小的排列，共做了六次比较，如果我们仍然采取那种方法对1000个数进行排序的话，那么就要进行499500次的比较。若一条一条地将程序写出来，那将是一项很艰巨的工作。怎么

办呢？我们用数组和下标变量来试一试。

首先要给1000个变量赋值，如果我们仍采用初级程序设计所采取的赋值方法，就过于麻烦了。怎样才能简便呢？我们知道，数组元素中，下标是有一定顺序的，根据这一特点，我们可以采取循环赋值的方法。其次是排序，我们用双重循环来实现。先看程序：

```
10 DIM A(1000)
20 FOR I=1 TO 1000
30 INPUT A(I)
40 NEXT I
50 FOR I=1 TO 999
60 FOR J=I+1 TO 1000
70 IF A(I)>=A(J) THEN 90
80 M=A(I): A(I)=A(J): A(J)=M
90 NEXT J
100 NEXT I
110 FOR I=1 TO 1000
120 PRINT I; " "; A(I)
130 NEXT I
140 END
```

这个程序中，20～40号语句是向1000个变量赋值，50～100号语句为从大到小排序，110～140是将1000个变量按照从大到小的顺序输出。

解释一下排序的过程：二重循环中，当外循环变量I取

某一值时，例如 I=1，A(1) 与内循环变量为 J=I+1 到1000的各变量值 A(2)～A(1000) 进行比较，若有大于A(1)的则变换位置，否则不变，使 A(1) 里存放的数据始终为最大。当内循环结束，便选出一个最大数A(1)，然后A(2) 再依次与剩余变量比较，取出第二大数 A(2)，一直这样做下去，便会将1000个变量按照由大到小的顺序排列好。

这儿，仅仅数十条语句便可将1000个变量排列出来。由此看来，引入数组和下标变量的概念能帮助我们解决一些棘手的问题，从而使程序更简明、更清晰。

[例 1] 计算一元多项式的值，即：

$y_n(x)=a_nx^n+a_{n-1}x^{n-1}+\cdots+a_0$

在初级程序设计里，我们仅仅能对某一个较低次数的具体的一个多项式进行处理，计算几个自变量所对应的多项式的值，而对要编制一个通用的、能计算任意多项式值的表达式，就显得无能为力了；数组的引入使这个问题得到解决。

对于多项式： $y_n(x)=a_nx^n+a_{n-1}x^{n-1}+\cdots+a_1x+a_0$，

经过提取公因式：

$y_n(x)=((\cdots((0*x+a_n)*x+a_{n-1})*x+\cdots)*x+a_1)*x+a_0$

整个运算被分解为下列各部分：

$y_0=0*x+a_n$

$y_1=y_0*x+a_{n-1}$

$y_i=y_{i-1}*x+a_{n-i}$

$y_n=y_{n-1}*x+a_{n-n}=y_{n-1}*x+a_0$

整个问题由一个表达式 Y=Y*X+A 进行若干次迭代而完成。我们用A(I)来表示一元多项式第I次项的系数，N表示X的方次，则程序如下：

```
10 READ N
20 DIM A(N)
30 FOR I=N TO 0 STEP -1
40 READ A(I)
50 NEXT I
60 INPUT "请输入X的值"; X
70 IF X=-1 THEN 160
80 Y=0
90 FOR I=N TO 0 STEP -1
100 Y=Y*X+A(I)
110 NEXT I
120 PRINT "X="; X, "Y="; Y
130 INPUT "还继续运算吗?(Y/N)"; S$
140 IF S$="Y" THEN GOTO 60
150 DATA……, -1
160 END
```

这样当我们依次输入多项式的次数、系数和 $x$ 值后，便可计算此多项式的值了。

[例 2] 选拔运动员。有15名长跑运动员，参加1500米长跑。当我们将他们的姓名和成绩键入计算机后，要求按名次打印出他们的姓名、成绩。

前面我们已经将1000个数据按照由大到小的顺序进行了排序，因而这个问题看起来就比较简单了。只需要在排序过程交换变量时，将存放姓名的字符串数组元素，一起交换位置便可。程序如下：

```
10 DIM A(15)，A$(15)
20 PR#1
50 FOR I=1 TO 15
40 INPUT“请输入运动员姓名”；A$(I)
50 INPUT“请输入运动员成绩”；A(I)
60 NEXT I
70 FOR I=1 TO 14
80 FOR J=I+1 TO 15
90 IF A(I)<=A(J) THEN GOTO 120
100 M=A(I)：A(I)=A(J)：A(J)=M
110 M$=A$(I)：A$(I)=A$(J)：A$(J)=M$
120 NEXT J
130 NEXT I
140 FOR I=1 TO 15
150 PRINT“第”；I；“名”，A$(I)，A(I)；“秒”
160 NEXT I
170 PR#0
180 END
```

## 60.打印杨辉三角形

### ——介绍二维数组和双下标变量

[例 1]

我们先来看表4-6，这是常见的学生成绩统计表。

表 4-6

| 学号 | 语文 | 数学 | 政治 | 物理 | 化学 |
|---|---|---|---|---|---|
| 1 | 90 | 85 | 88 | 92 | 94 |
| 2 | 80 | 90 | 93 | 89 | 74 |
| 3 | 85 | 87 | 92 | 91 | 86 |
| ⋮ | | | | | |
| ⋮ | | | | | |
| ⋮ | | | | | |
| 40 | 86 | 72 | 68 | 76 | 74 |

这个表格我们称为二维表格，它是由40行(除表头外)，六列组成的；表格中的任何一项，都代表了某学号学生某门功课的成绩。例如第二行第三列，它代表了学号为2的同学的数学成绩，它的值是90，也就是为90分。假如我们将这样一个表格作为一个数组处理的话，这个数组中的各个元素，必需要有两个下标：一个下标代表它在表格中所处的行号；一个下标代表它在表格中所处的列号。假如我们为这个数组取名为A，那么2号同学的数学成绩就是 A(1，2)——

这就是双下标变量的表示。和单下标变量的规定一样，它的两个下标也可以是变量和表达式，但事先一定要有确定的值，如A(I，J)、A(K+M，L+N)等，两下标之间要用逗号分开。同样，下标的值不能为负数，若带有小数，计算机自动取整。

表 4-6 作为数组，就称为二维数组，取名为A；其中每一个双下标变量A(0，0)，A(0，1)等，被称为其数组元素。数组元素名和它的值是两回事，例如2号同学数学成绩我们存放在A(1，2)里，它的值为90。

二维数组和一维数组一样，为了使计算机预留存贮空间，需要在使用之前，即程序的开头加以说明。

二维数组说明语句格式如下：

| 行号 DIM 数组名(表达式1，表达式2) |
|---|

其中：表达式1给出第一个下标的上界；

表达式2给出第二个下标的上界。

例如,上面的二维数组A,其说明语句就应写为：10 DIM A(39，5)；代表了它有40行，6列，当计算机接受此信息后，便会留出40×6＝240个存贮单元。下面我们一起来完成这张表格的打印工作。

①分析：首先要将A数组中各元素的值赋给各下标变量，一维数组的赋值方法是采用单重循环来实现的，那么二维数组就要采用双重循环来实现。同样，输出打印也要采用

双重循环。

②程序如下：

```
10 DIM A(40, 6)
20 FOR I=1 TO 40
30 FOR J=1 TO 6
40 READ A(I, J)
50 NEXT J
60 NEXT I
70 PRINT "学号"; "  "; "语文"; "  "; "数学";
   "  "; "政治"; "  "; "物理"; "  "; "化学"
80 FOR I=1 TO 40
90 FOR J=1 TO 6
100 PRINT A(I, J); "    ";
110 NEXT J
120 PRINT
130 NEXT I
140 DATA 1, 90, 85, 88, 92, 94
150 DATA 2, 80, 90, 93, 89, 76
        ⋮
        ⋮
1000 END
```

在这个程序里，要特别注意二维数组在内存中的排列顺序。若忽略了这个问题，就会使数据的输入输出格式达不到预定效果。在这个程序里，外循环变量 I 代表了第 1 个下

标，从1到40，实际含意为第I个学生，内循环变量J代表了第二个下标，实际含意为第J门功课。在输入数据时，输入的顺序是一个学生的信息读完再读下一个。

[例 2] 打印杨辉三角形。

在数学里讲到，杨辉三角形的每一行是 $(x+y)^n$ 的展开式的各项系数。如第一行的1就是 $(x+y)^0$ 的系数；第二行1，1是 $(x+y)^1$ 的系数；第三行的1，2，1是 $(x+y)^2$ 的展开式 $x^2+2xy+y^2$ 各项的系数；10行的杨辉三角形如图4-16。

```
1
1  1
1  2  1
1  3  3  1
1  4  6  4  1
1  5  10 10 5   1
1  6  15 20 15  6   1
1  7  21 35 35  21  7  1
1  8  28 56 70  56  28 8  1
1  9  36 84 126 126 84 36 9  1
```

图 4-16

从图4-16可以看出对角线和每一行的第一列都是1，其余各项是它的上一行中前一个元素和上一行的相应元素的元素之和。例如，第五行第三列的值是6，它是第四行第二列

3和第三列3之和。

因此，我们先将每行第一列和对角线元素赋1，然后，再赋其余元素，最后将输出杨辉三角形。

图4-17是程序的粗框图。对于一个较为复杂的程序，我们可以将程序的几个大部分分别画出框图，即称为粗框图，然后将每一部分具体程序执行路线细画出来。这样采取各个击破的方法，问题就会迎刃而解了。

下面我们将"计算其余元素"一框画出具体程序执行路线。

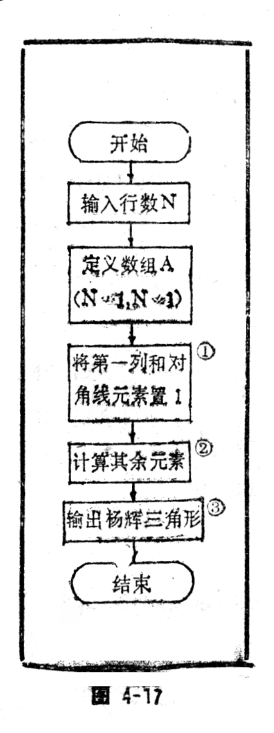



图 4-17

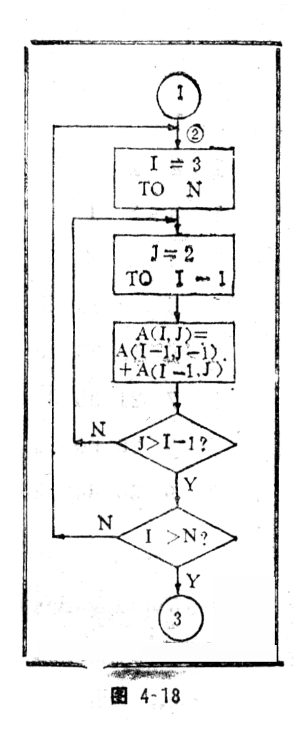



图 4-18

①分析：其余元素是一个二维的表，因此对于它们的赋值必定要用双重循环。外循环变量控制行数，从第三行开始到第$N$行；内循环变量控制列数。

②“计算其余元素”此部分的框图如图4-18（其它两部分省略了）。

●程序：

```
10 INPUT“请输入扬辉三角形的行数”；N
20 DIM A(N，N)
30 FOR I=1 TO N
40 A(I，1)=1
50 A(I，I)=1
60 NEXT I
70 FOR I=3 TO N
80 FOR J=2 TO I-1
90 A(I，J)=A(I-1，J-1)+A(I-1，J)
100 NEXT J
110 NEXT I
120 FOR I=1 TO N
130 FOR J=1 TO I
140 PRINT A(I，J)；“ ”；
150 NEXT J
160 PRINT
170 NEXT I
180 PRINT
```

```
190 END
```

我们在计算机上运行此程序，只要输入行数，便会输出相应的杨辉三角形，若想打印在纸上，可在程序前面加上PR#1。

## 61.哥德巴赫猜想在学习机上得到验证

——介绍子程序(GOSUB/RETURN语句)

前面我们曾经打印输出学生成绩表，为了整齐美观，我们准备在造表时行与行之间都要加上间隔线，现在让我们把这个程序的输出部分拿来修改加工，实现这个功能：

```
5  DIM A(40, 6)
10 PRINT TAB(10); "学生成绩表"          表头
20 FOR I=1 TO 80    ⎫
30 PRINT "-";       ⎬ 打印一条横线
40 NEXT I           ⎭
50 PRINT "学号"; " "; "语文"; " "; "数
   学"; " "; "政治"; " "; "物理"; " ";
   "化学"
60 FOR I=1 TO 80
70 PRINT "-";       ⎫ 打印一横线
80 NEXT I           ⎭
90 FOR I=1 TO 40
```

```
100 FOR J=1 TO 6
110 PRINT A(I, J); "  ",
120 NEXT J
130 PRINT
140 FOR K=1 TO 80   }
150 PRINT "-";      } 打印一横线
160 NEXT K          }
170 PRINT
180 NEXT I
190 END
```

从上面程序可以看出，多次运用了打印横线这一段程序，每次用到都要重写一遍。为了减少程序段书写次数，同时为了节省内存单元，可以引进子程序，即将多次用到的程序段作为子程序，可被返复调用；能够调用子程序而本身不被调用的程序，称为主程序。

主程序调用子程序要用转子语句，而当子程序执行完后要返回主程序（用返回语句）。

格式为：

| 主程序 | GOSUB 子程序开始行号 |
|---|---|
| 子程序 | RETURN |

其功能是当主程序执行到GOSUB 语句时，就转入执行GOSUB 后面行号为起始行号的子程序。当子程序执行完

后，由返回语句RETURN转入调用子程序的 GOSUB 的下一行号语句去继续执行。子程序是主程序以外的一个程序，因此子程序一般编号在主程序结束之后的程序中。子程序还可以调用其他子程序，称为子程序嵌套。

现在我们将打印输出横线的一段程序作为子程序，写在主程序之后：

```
10 PRINT TAB(10)；“学生成绩表”
20 GOSUB 300
30 PRINT “学号”；“  ”；“语文”；“  ”；“数
   学”；“  ”；“政治”；“  ”；“物理”；“  ”；
   “化学”
40 GOSUB 300
50 FOR I=0 TO 39
60 FOR J=0 TO 5
70 PRINT A(I，J)；“  ”；
80 NEXT J
90 PRINT
100 GOSUB 300
110 NEXT I
120 END
300 FOR K=1 TO 80
310 PRINT “-”；
320 NEXT K
330 RETURN
```

它的执行路线如图4-19。

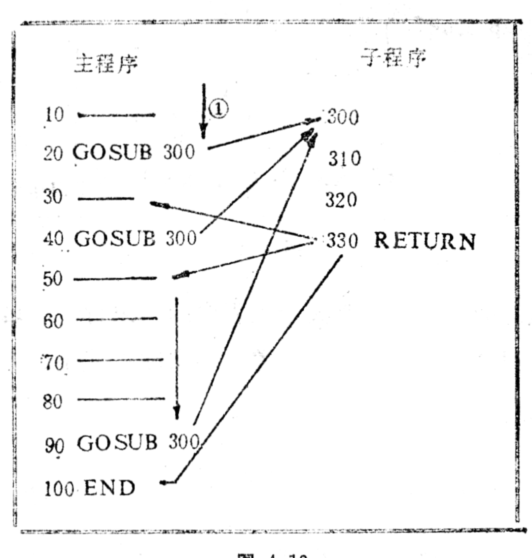



图 4-19

首先从标号10的主程序开始执行，20语句遇到 GOSUB 300时，转去执行300、310、320，执行到330号语句RETURN时，返回主程序GOSUB 的下一语句（30号语句）再顺序执行。

从上面修改后的程序可以看出，由于采用了子程序，去掉许多重复出现的冗余程序，使程序更加精练，更加模块化，从而显得结构清晰、易编、易读、易于检查错误。

子程序可以多层嵌套，即子程序可以有更小的子程序。其执行过程如图4-20。

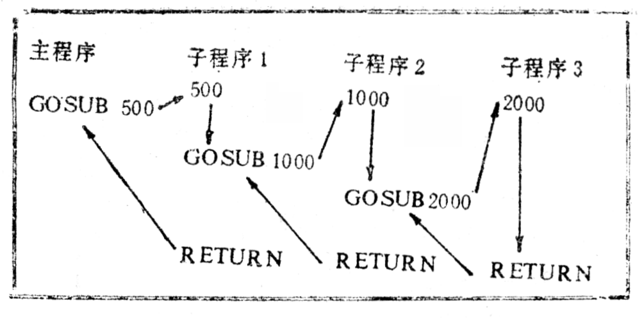



图 4-20

子程序的起始行号要大于主程序的结束行号，这是为了避免当主程序运行完毕后，接着运行实际上并不打算再去运行的子程序。

［例 1］ 计算组合数 $C_n^k\left(C_n^k=\dfrac{n!}{k!(n-k)!}\right)$。

在这个式子里，三次用到阶乘，因此，将计算阶乘作为子程序处理。程序如下：

```
10 INPUT “请输入N，K”;N, K
20 S=N
30 GOSUB 300
40 D1=J
50 S=K
60 GOSUB 300
70 D2=J
80 S=N-K
90 GOSUB 300
```

```
100 D3=J
110 D=D1/(D2*D3)
120 PRINT "C("; N; ","; K; ")="; D
130 END
300 J=1
310 FOR I=1 TO S
320 J=J*I
330 NEXT I
360 RETURN
RUN ↵
请输入N，K？5，3
C(5，3)=10
RUN ↵
请输入N，K？10，6
C(10，6)=210
```

在这个程序里，多次出现赋值语句，大家可以分析一下它们的作用。其一为设置调用参数，其二为接收返回参数。

为什么要考虑调用参数和返回参数呢？这是因为子程序一般都是做类同的工作，而不是做完全相同的工作。例如在这个程序中，尽管都是求阶乘，但三次阶乘的参数各不相同，因此在调用之前，须将参数赋于子程序，而所谓返回参数，就是将子程序的结果返回到主程序，其中S＝N，S＝N－K，S＝K均为设置调用参数，而 $D_1$＝J，$D_2$＝J，$D_3$＝J却为接收返回参数。

［例 2］ **编写验证500～100之间的偶数都可以表示为二个素数之和的形式（哥德巴赫猜想）。**

素数的定义和求法前面我们已经讲过，现在要验证，偶数＝素数＋素数。我们首先取一个小的素数，从已知偶数中减去，验证剩余的一个数是否为素数。若是，我们用 L＝1 来表示，工作完成；若不是，再将第一个素数值增大，从已知偶数中减去，再次验证剩余的一个数是否为素数。此项工作一直做下去，直到找到另一个素数为止。

我们将判断素数的过程作为子程序来处理。框图如图4-21。

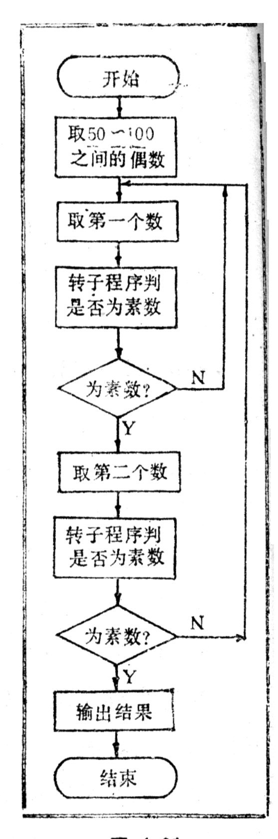


图 4-21

程序如下：

```
10 FOR M=50 TO 100
   STEP 2
20 W=1
30 W=W+1
40 P=W
50 GOSUB 300
60 IF S=0 THEN 30
70 Y=M-W
```

```
80 P=Y
90 GOSUB 300
100 IF S=0 THEN 30
110 PRINT M; "="; W; "+"; Y
120 NEXT M
130 END
300 S=0
310 FOR I=2 TO SQR(P)
320 IF P/I=INT(P/I) THEN 350
330 NEXT I
340 S=1
350 RETURN
RUN
50=3+47
52=5+47
54=7+47
56=3+53
58=5+53
60=7+53
62=3+59
64=3+61
    ⋮
```

经过验证 50～100 之间的偶数均可以表示为二个素数之和。

## 62.函数家族

——介绍标准函数和自定义函数

BASIC语言中函数主要分为两大类：一类是标准函数，它是由 BASIC 系统提供的内部函数；另一类是自定义函数，它是由用户自己根据需要而定义的函数。

前面我们已经接触过不少函数，如取整函数 INT(X)、平方根函数SQR(X)、三角函数SIN(X)和随机函数 RND(X)等，均属于标准函数。标准函数包括三角函数、算术函数、字符串函数和其它一些函数几类。

(1) 标准函数是由 BASIC 语言解释程序事先已经编制好的函数，因而我们在使用时可以直接调用。

①三角函数和算术函数

这两类函数比较简单，使用时只要写出函数名和自变量就可以了。现将函数和功能列于表4-7。

②字符串函数

A．LEN 是用来测量字符串长度的。字符串长度指字符串内字符的个数，其中包括空格、LEN(X＄) 就是用来测量字符串X＄中所包含的字符的总数。

例如：
```
10 A$ = "ABCDEFG"
20 PRINT LEN(A$)
30 END
RUN ↵
7
```

表 4·7

| 函　　数 | BASIC符号 | 含　　义 | 说明和例子 |
|---|---|---|---|
| 正弦函数<br>余弦函数<br>正切函数<br>反正切函数 | SIN(X)<br>COS(X)<br>TAN(X)<br>ANT(X) | 求X的正弦值<br>求X的余弦值<br>求X的正切值<br>求X的反正切值 | 三角函数中的变量$x$，一律以弧度为单位<br>例如：SIN45°应写为<br>SIN(45*3.1416/180) |
| 平方根函数 | SQR(X) | 求X的平方根 | 自变量X必须大于或等于零 |
| 指数函数 | EXP(X) | 求$e^X$值 | (e=2.7182818) |
| 对数函数（以e为底） | LOG(X) | 求$\log_{10}X$的值 | 例LOG(3)=0.47712126，x必须大于零 |
| 绝对值函数 | ABS(X) | 求X的绝对值 | 例 ABS(-5.5)=5.5 |

B. LEFT$，MID$，RIGHT$是用来取子字符串的。有时我们需要处理字符串中某一部分，这一部分我们就叫做字符串的子字符串，如 A$="ABCDE"，则字符串"A"，"AB"，"ABC"，"D"，"CDE"，"ABCD"，"BCDE"等都是"ABCDE"的子字符串。而LEFT$、MID$、RIGHT$三个函数就是用来从字符串中取出一部分字母组成新的字符串。

LEFT$(X$，N）是从X$字符串中左边的第一个字符开始取起，共取N个字符。如：

10 A$= "BASIC"

```
20 B$=LEFT$(A$, 3)
30 PRINT B$
40 END
RUN ↵
BAS
```

MID$(X$, P, N) 是从字符串X$中第P个字符开始取，共取N个，也可以写成MID(X$, P) 是从字符串X$中第P个字符开始取，一直到字符串末尾。如：

```
10 A$= "BASIC"
20 B$=MID$(A$, 3, 2)
30 PRINT B$
40 END
RUN ↵
SI
```

RIGHT$(X$, N) 是取出字符串X$中最右边的N个字符，如：

```
10 A$= "BASIC"
20 B$=RIGHT$(A$, 3)
30 PRINT B$
40 END
RUN ↵
SIC
```

C．字符串是可以连接的。两个或几个字符串可以用"+"号连接起来。如

```
10 A$= "I AM A"
20 B$= "STUDENT"
30 C$=A$+B$
40 PRINT C$
50 END
RUN
I AM A STUDENT
```

D. STR$和VAL 是数字变换器。STR$ 和 VAL分别可以将一个数学表达式的值转换成字符串或将字符串转换成数学表达式，如：

```
10 A=12: B=3: C=A+B
20 C$=STR$(A)+STR$(B)
30 PRINT C, C$
40 END
RUN
15      123
```

其中15为数字，而123为字符串。又如：

```
10 A$= "123456"
20 B=VAL(A$)
30 C=654321+B
40 PRINT A$
50 PRINT B
60 PRINT C
70 END
```

```
RUN ↵
1 2 3 4 5 6          …… 不是数字，是字符串。
1 2 3 4 5 6          …… 这是数字。
7 7 7 7 7 7
```

E．ASC 和 CHR＄是亲戚。ASC(X＄）是将字符串X＄中第一个字符转换成 ASCII 码。ASCII 是 American Standard code for Information Interchange（美国信息交换标准代码）的缩写。计算机用的全部字符都是建立在ASCII码的基础上的，字符的ASCII代码最大是255。如：

```
10 PRINT ASC（"A"）；
20 PRINT ASC（"ABC"）；
30 T＄＝"Z"
40 PRINT ASC(T＄)
50 END
RUN ↵
65 65 90
```

而CHR＄（数学表达式)却是将数学表达式值的ASCII码转换成它所代表的字符，如：

```
10 PRINT "CHR＄(65)＝"；CHR＄(65)
20 PRINT "CHR＄(90)＝"；CHR＄(90)
30 END
RUN ↵
CHR＄(65)＝A
CHR＄(90)＝Z
```

以上将字符串函数全部介绍完了。其它一些函数大部分前面已经作了介绍，如取整函数 INT(X)、打印格式 函 数 TAB(X)和随机函数 RND(X) 这里就不再重复了，只有符号函数SGN(X)还没有和大家见过面，下面就来熟悉一下。

SGN(X)是取X的符号：当 X>0时，则为 1；当 X=0 时，则为 0；当X<0时，则为 -1。如

```
10 PRINT SGN(500)
20 PRINT SGN(0)
30 PRINT SGN(-3.3)
40 END
RUN ↵
 1
 0
-1
```

(2) 自定义函数——DEF FN

前面已经介绍了各种标准函数，在编程序的过程中，有时需要重复计算某一表达式的值，而引入自定义函数。

自定义函数的格式如下：

| DEF FN 函数名 (X)=表达式 |
|---|

其中：DEF 是定义符；

FN 是函数标记；

函数名可以是A～Z中任一个字母；

括号中的X是形式变量；

表达式是含有X的BASIC表达式，也就是所要定义的函数式。

这样，便可以用等号左边的式子来代替等号右边的表达式。

[例] 已知函数 $Y=5X^3+4X^2+3X+1$，当X由1变化到10时，求Y的值。

程序如下：

```
10 DEF FNY(X)=5*X↑3+4*X↑2+3*X+1
20 FOR I=1 TO 10
30 PRINT FNY(I)
40 NEXT I
50 END
```

首先将 $5X^3+4X^2+3X+1$ 定义为FNY(X)，其中X为形式变量，当程序执行到40号语句时，调用了此函数，但自变量换成了 I，根据 I 取不同的值代入表达式，从而计算出Y值来。

例 2] 计算在不同速度下的动能$\left(E=\frac{1}{2}mv^2\right)$。

程序如下：

```
10 DEF FNE(V)=1/2*M*V↑2
20 M=5.5
30 V=3.5
40 PRINT FNE(V), FNE(V*2), FNE(V*3)
```

50 END

上面程序可以把速度为 V、2V、3V 的动能显示在屏幕上，当一个长的程序多次使用同一函数时，用自定义函数特别方便。

但是在使用自定义函数时，也必须遵循一些具体的规定：每条自定义函数只允许在程序中的一行内写完，如果过长或过于复杂，可以拆成几条自定义函数语句来定义。

## 63.学籍管理模型

——介绍结构化程序设计

模块结构化程序设计是目前提倡的程序设计方法，它是将比较复杂的问题化为许多独立而又易于解决的简单问题，将问题由繁到简地逐步解决。程序设计得层次清楚，易于阅读和修改，是解决复杂问题的一种有效方法。

本节以中学学生成绩管理为实例，当然，真正的学生成绩管理要复杂得多，但通过这个模型，学会运用已经学过的程序设计技巧，提高程序设计能力，更有效地解决实际问题。

总的原则是：首先搞清楚你所要设计的程序需要哪几方面的功能，从而确定划分为若干个模块；画出模块层次图；对每一个模块又要分析它需要哪几个功能，进而分成更小的模块。这就是模块结构化程序设计。

（1）分析

我们现在要设计的是某一个班的学生成绩管理档

案。

假设某班有N名同学（每位同学对应有学号，姓名），M门功课，要计算机计算全班同学的总平均成绩、某一科的平均成绩、每一个学生的平均成绩，要将某一门功课按分数段进行统计分析，要将全班同学按成绩排名次。当输入学科编号之后，计算机应自动查询此学科平均分及此学科成绩分析结果；当输入某学生学号之后，可查询此学生各科成绩、平均分数、名次，最后输出以下信息：

①按学号顺序打印输出全班各同学的姓名，各门功课成绩、平均分、名次；

②按名次打印输出全班各同学的学号、姓名、各门功课成绩、总平均成绩。

对于这样复杂的问题，在编程序之前要仔细分析，然后分出模块并画出模块图。

根据需要的功能大致分为以下6个模块：

A.建档模块：功能是接收信息。

B.计算模块：主要是计算全班总平均分，某一科的平均分，每个学生的平均分。

C.统计分析模块：功能是将某科考试成绩划分为分数段，进行分析统计。

D.排序模块：其功能是将学生成绩按由高到低的次序进行排列。

E.查询模块：a.查询班级总平均分；b.查询某学科平均分；c.查询某科成绩分析结果；d.查询某学生成绩、平均

分、名次。

F.打印模块：a.按学号打印成绩、平均分、名次；b.按名次打印成绩、学号、平均分。

（2）框图

①模块层次如图4-22。

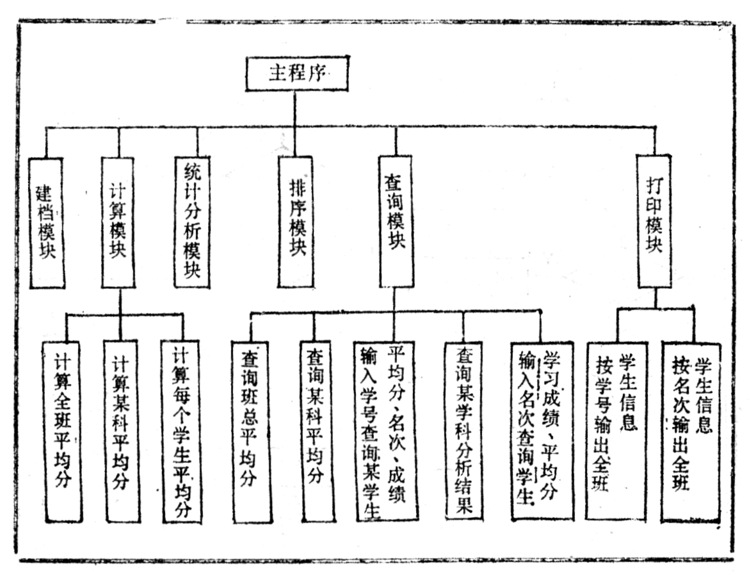



图 4-22

②主程序模块框图如图4-23。

③计算模块框图如图4-24。

④统计分析各科成绩模块框图如图4-25。

⑤排序模块框图如图4-26。

⑥查询学生学习成绩模块框图如（图4-27）。

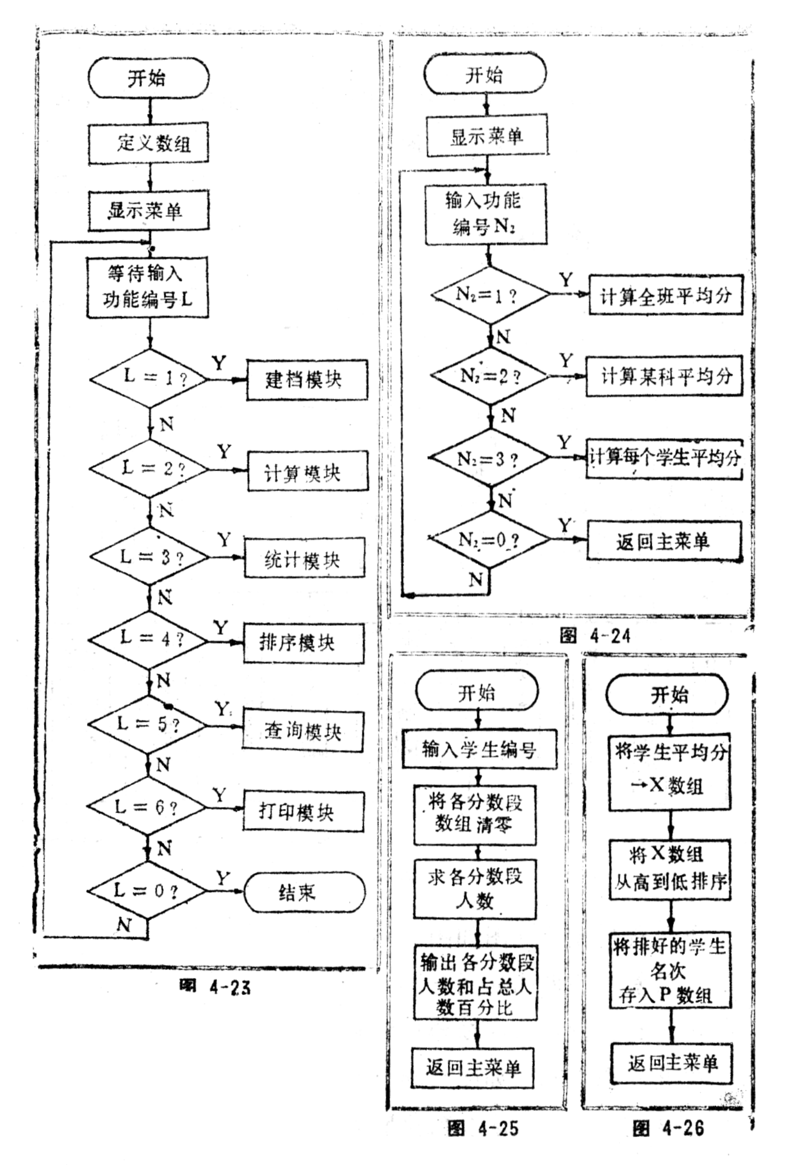



图 4-23

图 4-24

图 4-25

图 4-26

⑦打印输出模块框图如（图4-28）。

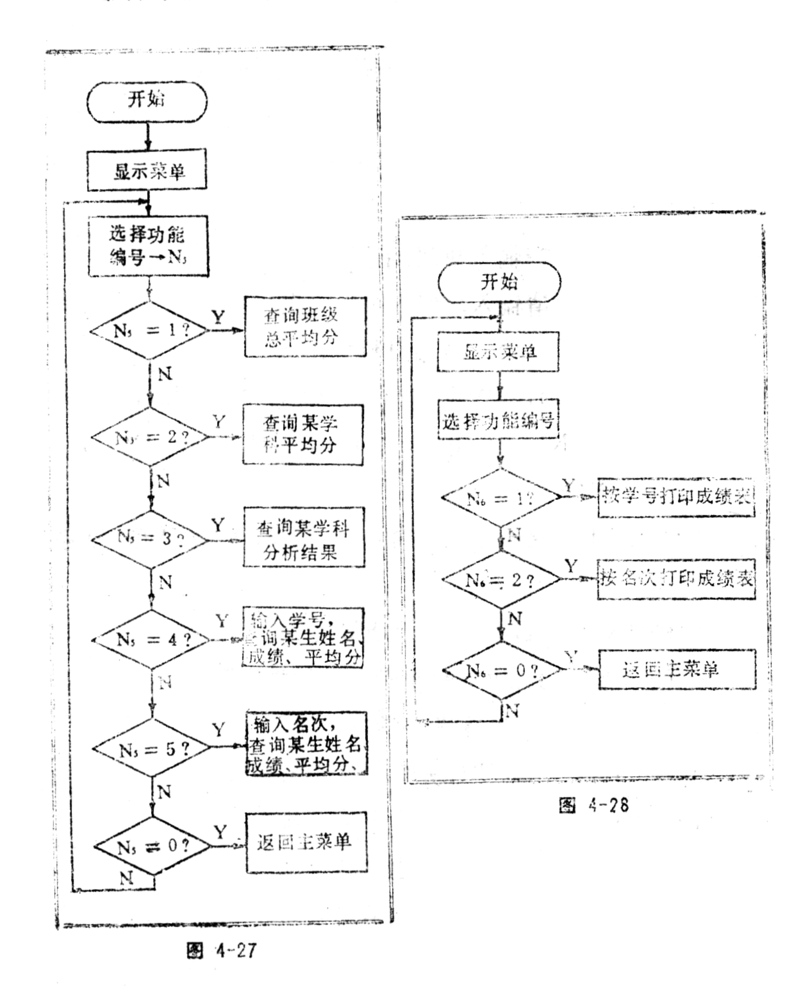



图 4-27

图 4-28

（3）数组定义一览表

K＄(M)—课程名；X(N)—中间数组；S(N)—学号，

XP(N)—某个学生平均分；A$(N)—姓名；D(I)— 某分数段人数；A(I,J)—某学生某科成绩；P(N)—名次；K(M)—某学科总分；KP(M)—某学科平均分。

(4) 程序编制

①主程序

```
10 HOME
20 INPUT“请输入班级编号”；N$
30 INPUT“请输入班级人数”；N
40 INPUT“请输入功课门数”；M
50 DIM S(N), A$(N), A(N, M), K$(M),
      X(N), XP(N), K(M), KP(M), D(11),
      P(N)
60 PRINT TAB(8)；N$；“班学生成绩管理系
   统”
70 PRINT TAB(12)；“功能表”
80 PRINT“1.建立学生学习成绩档案”
90 PRINT“2.计算学生学习成绩”
100 PRINT“3.统计分析学生学习成绩”
110 PRINT“4.将学生学习成绩排名次”
120 PRINT“5.查询学生学习成绩”
130 PRINT“6.打印学生学习成绩表”
140 INPUT“请键入你所需要的功能编号（若想退出
    操作请键入0）”；L
150 IF L=1 THEN GOTO 240
```

```
160 IF L=2 THEN GOTO 260
170 IF L=3 THEN GOTO 280
180 IF L=4 THEN GOTO 300
190 IF L=5 THEN GOTO 320
200 IF L=6 THEN GOTO 340
210 IF L=0 THEN GOTO 360
220 PRINT "你输入的功能编号错了，请 重 新输入"
230 GOTO 60
240 GOSUB 500      转建档模块
250 GOTO 60        返菜单
260 GOSUB 1000     转计算模块
270 GOTO 60        返主菜单
280 GOSUB 1800     转统计分析模块
290 GOTO 60        返主菜单
300 GOSUB 2000     转排序模块
310 GOTO 60        返主菜单
320 GOSUB 2500     转查讯模块
330 GOTO 60        返主菜单
340 GOSUB 3500     转打印输出模块
350 GOTO 60        返主菜单
360 END
```

很明显，主程序的作用是接收班级编号、班级人数、功课门数，然后定义数组，显示主菜单（功能表），接收操作者要执行的功能编号，然后执行此功能。

②建立学生学习成绩档案模块程序：

```
500 REM 建立学生学习成绩档案
510 HOME
520 PRINT "建立学生学习成绩档案"
530 PRINT "输入各门功课课程名"
540 FOR I=1 TO M
550 PRINT "第"; I; "门功课"; :INPUT K$(I)
560 NEXT I
570 PRINT"输入学号、姓名及各门功课成绩"
580 FOR I=1 TO N
590 INPUT "学号"; S(I)
600 INPUT"姓名";A$(I)
610 FOR J=1 TO M
620 PRINT K$(J); :INPUT A(I, J)
630 NEXT J
640 NEXT I
650 RETURN
```

其中500号语句为注释语句，为阅读程序带来方便。

在上面程序里，我们接收了这几个方面的数据：各门功课课程名，比如语文、数学等，并存入K$数组里。每个学生的学号存入S数组里，姓名存入A$数组，各门功课成绩存入A数组里。

③计算学生学习成绩模块程序：

```
1000 REM 计算学生学习成绩
```

```
1020 HOME
1030 PRINT TAB(8); “计算学生学习成绩”
1040 PRINT “1.计算全班总平均分”
1050 PRINT “2.计算某一学科总平均分”
1060 PRINT “3.计算全班每个学生平均分”
1070 PRINT “请你选择其中任一种操作（1/2/3），若
     要返回主菜单请键入0”
1080 INPUT N2
1090 IF N2=1 THEN GOTO 1140
1100 IF N2=2 THEN GOTO 1160
1110 IF N2=3 THEN GOTO 1180
1120 IF N2=0  THEN GOTO 1195
1130 PRINT “你的功能编号键错了，请再来一次”：
     GOTO 1070
1140 GOSUB 1200      ——转计算班级总平均分
1150 GOTO 1000       ——返计算模块功能表
1160 GOSUB 1400      ——转计算某一科平均分
1170 GOTO 1000       ——返计算模块功能表
1180 GOSUB 1600      ——转计算每个学生平均分
1190 GOTO 1000       ——返计算模块功能表
1195 RETURN          ——返主菜单（主程序功能表）
1200 REM 计算全班总平均分
1205 HOME
1210 PRINT“计算全班总平均分”
```

```
1215 Z=0                      ——清总分单元
1220 FOR I=1 TO N
1230 FOR J=1 TO M
1240 Z=Z+A(I, J)              ——计算总分
1250 NEXT J
1260 NEXT I
1270 B2=INT(Z/(N*M))*10+0.5)/10
                              ——计算总平均分
1280 PRINT "全班总平均分是："；B2
1290 RETURN
```

这个程序比较简单，要说明的是，计算出的全班总平均分，是按照四舍五入后保留一位小数来处理的。

④计算某学科平均分：

```
1400 REM 计算某学科平均分
1410 HOME
1415 PRINT "本学期课程及编号如下："
1420 FOR I=1 TO M
1430 PRINT I; "  "; K$(I),
                              ——显示全部学科及编号
1440 NEXT I
1450 PRINT
1460 INPUT "请输入要计算平均分的学科编号"，J
1470 K(J)=0                   ——所选学科总分单元清零
1480 FOR I=1 TO N
```

```
1490 K(J)=K(J)+A(I, J)     ——计算所选学科总分
1500 NEXT I
1510 KP(J)=INT(K(J)/N×10+0.5)/10
                              ——计算所选学科平均分
1520 PRINT  K$(J); "平均分为"; KP(J)
1530 RETURN
```

⑤计算每个学生平均成绩：

```
1600 REM 计算每个学生平均成绩
1610 HOME
1620 PRINT "正在计算每个学生平均分"
1630 FOR I=1 TO N
1640 X(I)=0              ——将每个学生总分单元清零
1650 FOR J=1 TO M
1660 X(I)=X(I)+A(I, J)      ——求每个学生总分
1670 NEXT J
1680 XP(I)=INT(X(I)/M*10+0.5)/10
                                  ——求每个学生平均分
1690 NEXT I
1700 RETURN
```

⑤统计分析某学科成绩模块程序：

```
1800 REM 统计分析某学科成绩
1810 HOME
1815 PRINT "本学期所学课程及编号如下："
1820 FOR I=1 TO M
```

```
1830 PRINT I , K$(I)
1840 NEXT I
1850 PRINT
1860 INPUT "请输入要统计分析的学科编号" ; J
1870 FOR I=0 TO 10
1880 D(I)=0        ——将本科每分数段人数单元清零
1890 NEXT I
1900 FOR I=1 TO N
1905 X=INT(A(I, J)/10) ——}
                          } 计算每分数段人数
1910 D(X)=D(X)+1       ——}
1912 NEXT I
1915 PRINT "100分的"; TAB(10); D(10); "人";
     TAB(20); "占总人数";INT(D(10)/N*10000)
     /100; "%" ——输出100分的占总人数的百分数
1920 FOR K=9 TO 0 STEP -1
1925 PRINT K*10; "分—" ; K*10+9; "分";
     TAB(10); D(K); "人" ;
1930 PRINT TAB(20); "占总人数的" ;
     INT(D(K) / N*10000) /100; "%"
1940 NEXT K——输出各分数段人数占总人数百分数
1950 RETURN
```

⑥排序模块程序:

```
2000 REM 排序
```

```
2010 PRINT "正在排序"
2020 FOR I=1 TO N
2030 X(I)=XP(I)    ——将每个学生平均分送中间数
                     组
2040 NEXT I
2050 FOR I=1 TO N-1
2060 FOR J=I+1 TO N
2070 IF X(I)>=X(J) THEN 2110
2080 T=X(I)
2090 X(I)=X(J)
2100 X(J)=T
2110 NEXT J
2120 NEXT I      ——以上为从高到低排序
2130 FOR I=1 TO N
2140 FOR J=1 TO N
2150 IF XP(I)<>X(J) THEN 2180
2160 P(I)=J
2170 GOTO 2190
2180 NEXT J
2190 NEXT I      ——以上是将排好的名次存入P数组
2200 RETURN
```

⑦查询学生学习成绩模块程序：

```
2500 REM 查询学生学习成绩
2510 HOME
```

```
2520 PRINT TAB(8); "查询学生学习成绩"
2530 PRINT "1. 查询班级总平均分"
2540 PRINT "2. 查询某学科平均分"
2550 PRINT "3. 查询某学科成绩分析结果"
2560 PRINT "4. 输入学号查询此学生姓名、各科成
     绩平均分、名次"
2570 PRINT "5. 输入名次,查询此学生学号、姓名、
     各科成绩、平均分"
2580 PRINT "请选择以上功能编号,若要退出,返主
     菜单,请键入0";
2590 INPUT N5
2600 IF N5=1 THEN GOTO 2660
2610 IF N5=2 THEN GOTO 2680
2620 IF N5=3 THEN GOTO 2700
2630 IF N5=4 THEN GOTO 2720
2640 IF N5=5 THEN GOTO 2740
2650 IF N5=0 THEN GOTO 2760
2660 GOSUB 1200          ——转计算班平均分
2670 GOTO 2510
2680 GOSUB 1400          ——转计算某学科平均分
2690 GOTO 2510
2700 GOSUB 1800          ——转某学科计算分析结果
2710 GOTO 2510
2720 GOSUB 2800          ——转按学号查询
```

```
2730 GOTO 2510
2740 GOSUB 3200          ——转按名次查询
2750 GOTO 2510
2760 RETURN              ——返主菜单
2800 REM 按学号查询
2810 HOME
2820 INPUT“请输入所要查询的学生学号”；K
2830 FOR I=1 TO N
2840 IF K=S(I) THEN GOTO 2860
2850 NEXT I
2860 PRINT “姓名：”;A$(I)
2870 PRINT“各科成绩为：”
2880 FOR K=1 TO M
2890 PRINT K$(K)； “:”； A(I, K)
2900 NEXT K
2910 PRINT “平均分：”； XP(I)
3000 PRINT“名次：”； P(I)
3010 RETURN
3200 REM 按名次查询
3210 HOME
3220 INPUT“请输入所要查询的名次”；K
3230 FOR I=1 TO N
3240 IF K=P(I) THEN 3260
3250 NEXT I
```

```
3260 PRINT “学号：”；S(I)
3270 PRINT “姓名：”；A$(I)
3280 PRINT “各科成绩为：”
3290 FOR K=1 TO M
3300 PRINT K$(K)； “:”；A(I, K)
3310 NEXT K
3320 PRINT “平均分：”；XP(I)
3330 RETURN
```

⑧打印输出模块程序

```
3500 REM 打印输出学生成绩
3510 HOME
3530 PRINT TAB(8)； “打印输出学生成绩”
3540 PRINT “1．按学号打印输出成绩表”
3550 PRINT “2．按名次打印输出成绩表”
3560 INPUT “请输入功能编号”；若想退出请键入
     0”；N6
3570 IF N6=1 THEN GOTO 3600
3580 IF N6=2 THEN GOTO 3620
3590 IF N6=0 THEN GOTO 3640
3600 GOSUB 3700
3610 GOTO 3510
3620 GOSUB 3900
3630 GOTO 3510
3640 RETURN
```

```
3700 REM 按学号打印输出学生成绩表
3710 HOME
3715 PR#1
3720 PRINT TAB(16); "学生学习成绩表"
3730 PRINT TAB(3); "学号" ;TAB(8);"姓名";
3740 PRINT TAB(13); "平均" ; TAB(21);名次;
3750 FOR I=1 TO M
3760 PRINT TAB(5*I+21); K$(I)
3770 NEXT I
3780 FOR I=1 TO N
3790 PRINT TAB(3); S(I); TAB(8); A$(2);
3800 PRINT TAB(15); XP(I); TAB(21); P(I);
3810 FOR J=1 TO M
3820 PRINT TAB(5*J+21); A(I, J);
3830 NEXT J
3840 PRINT
3850 NEXT I
3860 RETURN
3900 REM 按名次打印输出学生成绩表
3910 HOME
3920 PRINT TAB(16); "学生学习成绩表"
3930 PRINT TAB(3);"名次"; TAB(8); "姓名"
3940 PRINT TAB(15);"学号"; TAB(21); "平均"
3950 FOR I=1 TO M
```

```
3960 PRINT TAB(5*I+21); K$(I)
3970 NEXT I
3980 FOR K=1 TO N
3990 FOR I=1 TO N
4000 IF K<>P(I) THEN 4070
4010 PRINT TAB(3); K; TAB(8); A$(I)
4020 PRINT TAB(15); S(I); TAB(21); KP(I)
4030 FOR J=1 TO M
4040 PRINT TAB(5*J+21); A(I, J);
4050 NEXT J
4060 PRINT
4070 NEXT I
4080 NEXT K
4090 RETURN
```

# 第五章　中华学习机使用指南

## 一、帮你组成一套实用的中华学习机系统

### 64.系统的配置

中华学习机系统部件很多，如主机、磁带机、显示器、打印机、软盘驱动器，但这些部件并非缺一不可，利用它的最大优点——积木式结构，我们可以根据自己的条件和使用的目的等来选择系统的配置。

目前，大多数家庭都有了电视机和录音机，再配上一台主机便构成了中华学习机的最小系统，利用最小系统可以编制程序、计算题目和作游戏等。

当然，如果想使用更多的成品软件来解决学习中的问题，用于辅助教学、开发学校管理系统等，那么最好是配置软盘驱动器。软磁盘容量大、读写速度快，而且方便。因此，目前许多软件都是存贮在软磁盘上，这样，只有配置软盘驱动器才能使中华学习机在更广泛的领域里发挥出更大的作用。

当然打印机也是很有用处的，在有条件的情况下配置它并不多余，它能将程序运行的结果、屏幕的画面和图形打印

在纸上，还可以输出报表，打印文稿等。

总之，在选择系统配置时，一要根据现有条件，二要根据使用目的，全面考虑。当然，也可以先配置最小系统，在有条件时或者需要时再逐步完善。

### 65．系统的连接

当确定了中华学习机的系统配置之后，便可以着手将它们连接在一起了。

（1）主机与显示器的连接

当使用电视机作为显示器时，连接方法如下：

①把电视机的天线插头拔掉；

②用视频电缆线一头插入主机后座标有“电视机”(TV)的插孔，一头插入电视机天线插孔；

③将电视机电源和主机电源插头分别插入电源插座；

④先开电视机电源开关，再开主机电源开关；

⑤调准电视机频道（一般在3或4频道）直到屏幕清晰为准。

当使用监视器作为显示器时，连接方法如下：

①用视频电缆线一头插入主机后座标有“监视器”的接口，一头插入监视器IN 插座；

②将主机及监视器的电源插头分别插入电源插座；

③先开监视器后开主机，屏幕上将出现清晰的“ZHONG HUA XUE XI JI”字样。

（2）主机与录音机的连接

将录音机电缆线的一端插入主机侧面的录音机接口，另一端的两个3.5mm插头分别插入录音机EAR插孔（连接主机IN）和MIC插孔(连接主机OUT)。

(3) 主机与软盘驱动器的连接

将主电路板软磁盘驱动器的一组插座插上相应的器件，即软盘驱动器带有一个20芯电缆将此电缆与主机座后面标有软盘驱动器的20芯插座相连。

(4) 主机和打印机的连接

先将打印机扩充板插入主机内相应的插座上，然后通过20芯电缆将扩充板上的插座与打印机插孔相连。

至此，中华学习机系统已经安装完毕，等待开机使用。

### 66.系统的启动

首先将主机和其它外部设备（电视机、录音机、打印机等）的电源接到220V电源插座上，当然如果有稳压电源更好，将稳压电源的输出调为220V，然后将主机和其它外部设备电源线接在稳压电源的输出端。

现在我们就可以开机了。这里，要分两种情况来介绍。

①没有软盘驱动器的情况下，先将外部设备的电源开关开启，然后在计算机主机的左后方找到主机电源开关，打开此开关，可听到扬声器发出“嘟”的一声响，屏幕显示中华学习机的汉语拼音“ZHONG HUA XUE XI JI ”字样，左下方显示“]”符号，我们称它为提示符，它告诉我们计算机处于何种工作状态（目前已自动进入BASIC状态），

屏幕上闪动着的"□"符号提示您下一个字符的显示位置，我们称它为光标。它的存在说明计算机目前可以接受键盘命令。

②系统配有软盘驱动器时，应首先将标有 DOS3.3的盘插入磁盘驱动器（注意在插入磁盘时要将有标签的一面朝上，缺口朝里）关上驱动器的门柄，然后打开外部设备开关，找到主机开关并打开，这时可听到扬声器发出"嘟"的一声响，屏幕显示"ZHONG HUA XUE XI JI"字样，说明计算机已经被启动，处于正常工作状态，如果没有听到声响，应立刻关闭电源进行检查，排除故障后再重新开机。

这时磁盘机上的指示灯发亮，并且可以听到马达转动的声音，直到磁盘机灯熄灭，屏幕将显示：

```
DOS VERSION 3.3                08/25/80
APPLE Ⅱ PLUS OR ROMCARD SYSTEM
MASTER
(LOADING INTEGER INTO LANGUAGE
CARD)
]□
```

此时，系统已在 BASIC 系统下，如果开机前没有将 DOS3.3 软磁盘装入，系统就会直接进入 APPLE BASIC 系统，如果需要装入DOS系统，此时可以将 DOS 盘插入磁盘机，并键入：

] PR#6 ↵

于是屏幕显示同上。

# 二、键 盘 操 作

## 67.中华学习机的键盘

中华学习机的键盘和主机是一体的（直接装在主机壳内），它们之间是通过一条26线扁平电缆相连接的，外形如图5-1。

从图5-1我们可以看出键盘共有69键，其中包括26个英文字母键、10个数字键和符号键，这些都属于一般常用键。除此之外，还有一些特殊键。

(1) | $F_1$ | $F_2$ | $F_3$ | $F_4$ | $F_5$ | 键

这五个键称为功能键，用户可以根据自己的需要定义它们的功能。例如：在后一部分讲到BASIC程序，当我们向内存输了一些程序之后想将全部程序列出来看一看，就要用到这样一个命令：LIST。这样一个个字母键入十分麻烦，我们就可以定义$F_1$为LIST，只要按下$F_1$键，屏幕上立即出现LIST，并将内存中的程序列于屏幕上，使操作既快又简便。当然在不同的系统中它有不同的用途，以下用到时再作介绍。

(2) | Ctrl | 键

称为控制键，当它与其它的键同时按下，可产生不同的控制作用。

(3) | Reset | 键

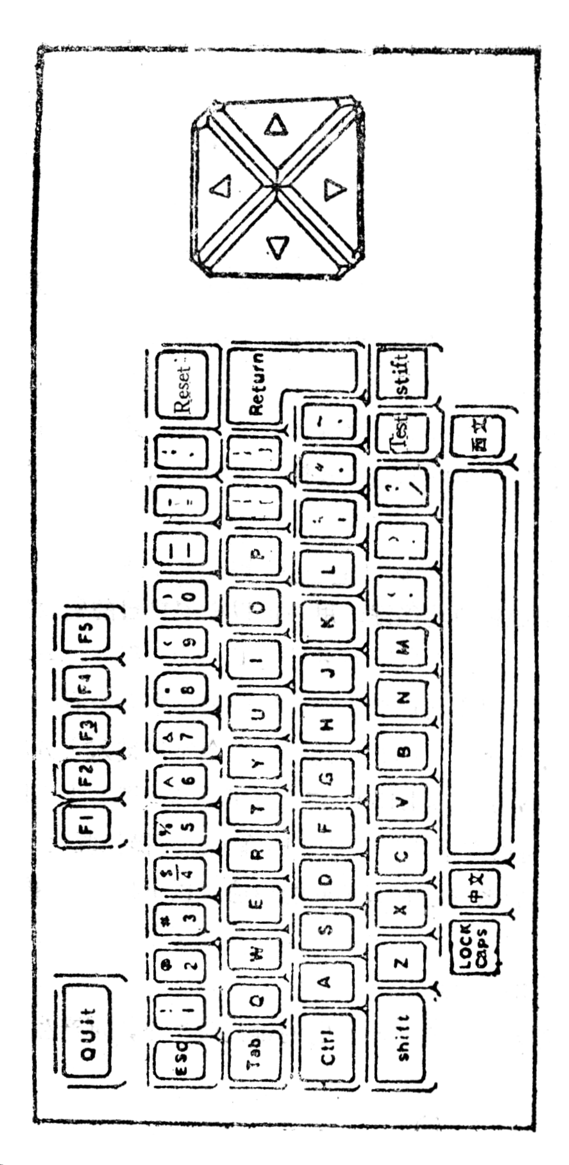

Quit
F1
F2
F3
F4
F5
Esc
Reset
Tab
Return
Ctrl
shift
Test
LOCK Caps
中文
西文
Q
W
E
R
T
Y
U
I
O
P
A
S
D
F
G
H
J
K
L
Z
X
C
V
B
N
M

图 5-1

用于系统复位。当通电后，系统运行中，由于操作失误或程序错误等原因，使系统不能正常执行（或者陷入死循环），需要系统重新启动：在按下 Ctrl 键的同时，按下 Reset 键，系统便重新启动。

(4) Return 键

这个键叫回车键，是键盘上最重要的键，也是用得最多的键，当键入一个命令后，只有按了 Return 键，计算机才真正地接受并执行。本书将此键用“↵”来代替。

(5) Shift 键

叫做变换键。键盘上，有的键标有两个字符，如标有数字“8”这个键，在其上方还标有符号“*”。如果我们要将“8”这个数输入计算机内，直接按该键即可；如果要输入乘号“*”，则需同时按该键和 Shift 键。因此，对于标有两个字符的键，要想输入上面的字符，就必须在按该键的同时，按下 Shift 键，所以此键又称为“上档键”。

(6) Lock Caps 键

是字母大小写的转换键。按下该键，键盘处于大写字母状态，此时字母键输入的信号，在计算机内或屏幕上均显示大写字母，再按一下该键，键盘则回到小写字母状态，这时凡输入的字母，均为小写字母且屏幕上显示的也是小写字母。

(7) 中文 和 西文 键

用于中西文方式的转换，通电后系统自动处于西文 BASIC 状态，若此时键入 中文 键，系统便进入中文 BASIC，若想再次进入西文 BASIC 可按 西文 键。

(8) |　　　　| 键

为空格键，在中文键和西文键之间，程序中要多次用到此键。

(9) | Test | 键

用于系统自检，这部分内容在以后章节里叙述。

(10) | ESC | 键

用于屏幕编辑，我们下节讲到。

## 68.键盘操作的要领

(1) 键盘操作的姿势

操作键盘要有正确的姿势，对于儿童来说就如同掌握写字要领一样重要，数十年后，键盘操作将与写字具有同等的地位。

正确的姿势应该是：头要正，背要直，两腿自然弯曲成90度，两脚平放地面与肩同宽，两大臂自然垂下，两小臂自然弯曲与大臂成90度；两腕与手臂成平行，两手如弓型，自然地放在键盘上，身体与键盘的位置可根据小臂的长短自由调整，一般在15～20厘米左右，只要大臂自然下垂而手指舒服地放在中排键上，这就说明身体离键位的距离合适了。

总之，在进行键盘操作之前要对姿势和距离进行自我检查，认为正确并感到舒适就可以进行键盘操作了。

(2) 击键的力度

首先手腕要保持平行，用大臂带动小臂做直上直下的活动，手指保持自然弯曲。手腕要绝对保持与小臂自然平行。

手指击键要迅速果断，每击一键手指要迅速弹回基本键位上，这样可以提高速度，而且省力，前键击后可利用弹起的力再击下一键。

### 69.屏幕编辑

屏幕编辑就是当我们通过键盘向计算机输入命令或者程序的同时，也显示在屏幕上，如果发现错误和格式不对时，就要进行修改，这个过程就称为屏幕编辑。

屏幕编辑要用到编辑键，以下介绍编辑键的用法。

首先讲一下最常用的两个编辑键。

① |◁|：按下此键，删除内存输入行的最末一个字符，并使光标在屏幕上左移一格。使用它就像我们通常擦字用的橡皮一样方便，输入过程中发现有错误就可用它来将其抹去，进行修改。

② |▷|：按下此键，可把光标处的字符输入到内存输入行的末尾，并使光标右移一格；这个键的使用一般要与其它编辑键配合使用。

下面介绍一套只改变屏幕上显示的字符，而不能改变内存输入行字符的编辑命令。

这套编辑命令是以 ESC 键打头的一套编辑命令。使用方法如下：按下 |ESC| 键后就进入编辑状态在该状态下，有11个键（A，B，C，D，E，F，@，I，J，K，M）——称为“ESCAPE码”，定义如下：

ESC A：光标右移一行；

ESC B：光标左移一行；

ESC C：光标下移一行；

ESC D：光标上移一行；

ESC E：从光标处开始清除到行尾；

ESC F：从光标处开始清除到屏幕尾；

ESC @：清屏幕，且光标回到屏幕的左上角；

ESC I：光标上移一行，仍处于 ESC 方式；

ESC △：同上；

ESC J：光标左移一格，仍处于 ESC 方式；

ESC ◁：同上；

ESC K：光标右移一格，仍处于 ESC方式；

ESC ▷：同上；

ESC M：光标下移一行，仍处于 ESC方式；

ESC ▽：同上。

由此可以看到 ESC与A，B，C，D，E，F，@配合，执行后自动退出ESCAPE 方式，而与I，J，K，M配合执行后仍处于 ESCAPE 方式，直到打入一个非 ESCAPE 码之后才退出 ESCAPE 方式。

## 三、汉字系统的使用

中华学习机与其它学习机相比，突出的优点是增加了汉字功能，这为我们的使用带来了很大的方便。

目前微型计算机上配有的汉字系统有许多种，实现方法

和输入方法也各不相同。中华学习机集中了它们的优点，在实现方法上采用了固化 ROM，就是把汉字的编码都做在几片非常小的集成电路块上，这样可不占用内存贮器空间，也就可以节省出许多内存贮器空间供我们利用。同时，这种实现方法查找汉字速度也非常快。

在输入方法上也作了一些改进，主要是针对学习机的使用对象是广大青少年和非计算机专业人员，因此要求输入方法简单、易学，采用了拼音输入法和区位码输入法。

拼音输入法是采用全拼音输入，即要输入某一个汉字，只要将此汉字的汉语拼音输进去，就可以将此汉字查找出来，对学过汉语拼音的人，特别是广大青少年就相当方便。

区位码输入法是采用国际 GB2381-80 区位方式输入汉字。要输入某一个汉字就要将它所对应的区位码输进去，很显然这需要记忆大量的数码。那么为什么要采用这种输入方法呢？这是因为区位码输入法查找速度快。我们适当记一些常用符号和汉字的区位码，可大大加快我们的输入速度。

### 70.汉字系统的启动与退出

（1）汉字系统的进入

当加电后系统自动处于西文BASIC状态，屏幕显示为低分辨率字符方式，即整个屏幕可显示24行，每行40个字符。此时键入“中文”键，则进入中文 BASIC，屏幕显示方式为高分辨率图形方式，全屏幕可显示每行17个汉字，共10行，此时下部第11行为状态行，状态行左端显示“西文”

两字，这是状态提示符，它表示系统处于中文BASIC下的西文字符输入状态。

（2）汉字系统的退出：

在中文状态下，直接键入“西文”键，便可退出中文状态，进入西文 BASIC 状态。

## 71.汉字的输入方法

进入汉字系统后，系统处于字母输入方式，这时通过键盘输入的是 ASCII 字符，通过按下“F1”～“F5”功能键可以选择不同的汉字输入方式。功能键的定义如下：

“F1”（CTRL—A）——字母方式；

“F2”（CTRL—L）——拼音输入方式；

“F3”（CTRL—W）——区位码输入方式；

“F4”（CTRL—T）——保留给用户扩充新的输入方案；

“F5”（CTRL—F）——保留给用户扩充新的输入方案。

（1）拼音输入法

在进入汉字系统后，按下“F2”键（或CTRL—L）后，屏幕提示行将显示“拼音两字”，表示进入拼音输入方式。

拼音输入方式使用26个字母键，大小写字母均可输入。

中华学习机采用全拼音输入，即输入汉字时，将该汉字的音节（声母和韵母）全部打入，计算机根据输入的音节在状态行提示出该音节的所有字，包括不同声调的字，操作者根据提示选择所需要的字。因此，只要学会全拼音输入方式，就

不用再记忆输入码了，这很有利于儿童掌握计算机。

其输入过程可以分以下几步：

A.键入声母。此时，屏幕状态行上显示出以该声母开头的音节中最常用的6个汉字，此时若其中有要输入的汉字，只要打入相应的序号（每个汉字之前都有序号）即可。

B.键入韵母。

C.向前或者向后寻找同音字。当键入韵母之后，屏幕状态行提示出6个同音节字，若这六个同音字中没有要输入的汉字，可键入 ⊡ 键，这时在屏幕状态行又会显示出下一幕六个同音节字，若还没有继续按 ⊡ 键，直到计算机发出"嘟"声，表示已全部显示完，当你要寻找前一幕汉字时，可敲 ⊡ 键，此时前一幕的六个汉字显示在状态行上。

D.键入所选择的序号。当状态行上出现要输入的汉字时，只要打入该汉字前面的序号，该汉字就显示在屏幕光标位置上，汉字内码输入到键盘输入缓冲区，连续打入序号，可连续输入汉字。

E.删除拼音提示。若输入时，有的拼音打错了，可直接按 ◁ 键，使状态行上最后一个拼音字母被删除，当状态行上的拼音都删除掉之后，再按 ◁ 键，则删除光标上的信息。

当需要退出拼音输入方式时，可在按下CTRL键的同时按下L键，屏幕下部状态行显示"西文"两字，此时即退出拼音输入方式。

(2) 区位码输入法

在汉字系统状态下，按“F3”键（或CTRL—W）后，屏幕状态提示行将出现“区位：”两字，表示进入区位码输入方式。

区位码输入方式是使用国家标准GB—2381—80区位方式输入汉字。在输入汉字时，仅使用0～9共10个数字键作为汉字的输入键，通过敲入四位数字键可以把国家标准中所规定的汉字和图形字符输入到计算机中，如国标中规定16区01位置上的汉字是“啊”，键入1601后，“啊”字就输入到计算机中了，即：

| 区位码 | 汉字 |
| --- | --- |
| 16（区码） 01（位码） | 啊 |

在使用区位码输入汉字时，除0～9数字键作为汉字键入码之外，键盘上的其它键均以原来的ASCII字符来处理。

在输入汉字过程中，如果有一输入码敲错，可用 |◁| 键来删除。如果所键入的区位码不在国标码所规定的范围之内，系统会发出蜂鸣声警告用户，拒绝接收所输入的区位码，等待用户重新键入或删除修改。

以上介绍的是两种汉字输入方式，中华学习机还有其它输入方式，这里就不再一一介绍了。

## 四、如何利用磁带存取信息

当把中华学习机作为家庭电脑来使用时，一般都是将录

音机（磁带机）作为外存贮器，它不但经济实惠，而且使用方便。为此，中华学习机增强了这方面的功能，以便使中华学习机更快地进入千家万户。

## 72.将BASIC程序存入磁带

当我们在主机上调试好一个BASIC 程序，可以记录在录音机的磁带上，等需要时可随时调出运行。将主机内存中的BASIC程序存入磁带的命令如下：

格式： SAVE [“文件名”]；

其中：文件名用方括号括起来，表示可以选择。

使用方法是在键盘上敲入SAVE和文件名之后，同时按下磁带机的放音键 PLAY 和录音键REC，几秒钟之后按下键盘上的回车键。在程序装入磁带的过程中，计算机会发出两个“嘟”声，在装入完毕后，屏幕上会显示 BASIC 提示符“]”，这时按下录音机的 STOP 键。若不敲入文件名，也可以将程序存入磁带，但假若再次将程序从磁带上装入内存时，就只能凭文件在磁带上的位置来装入文件了。

## 73.将磁带上的 BASIC 程序装入内存

将磁带上的BASIC程序装入内存的命令如下：

格式： LOAD [“文件名”]；

其中：文件名可以不给出。

使用方法是在主机键盘上敲入 LOAD 和文件名后，按下录音机的放音键PLAY，几秒钟后，按下键盘的回车键，

程序开始装入内存；在此过程中，机器会发出“嘟”的一声。当装入完毕后，屏幕上显示出文件名及BASIC提示符，如果装入的文件并非是你所要装入的文件时，屏幕上会提示已装入的文件名和如下信息：“HAS BEEN LOADED”，并继续装入后面的文件，直到装入LOAD语句所指定的文件为止；如果没有给出文件名，LOAD命令只将第一个文件调入内存，然后显示BASIC提示符。

### 74.将磁带中游戏软件装入内存并自动运行

当把中华学习机作为家庭电脑时，玩游戏便成为主要用途之一，因中华学习机与苹果机兼容，所以磁带游戏软件相当丰富。为此，专门增加了将录音机中游戏软件装入内存并自动运行的命令：

格式：PLAY。

使用方法是先将磁带退到开始位置，在键盘上打入PLAY命令，但注意在键入该命令后先别按回车键，在录音机上按下放音PLAY键，使磁带开始转动，再按下键盘上回车键（RETURN），这时屏幕上光标消失，大约15秒钟，会出现“PLEASE WAITING…”之类的信息，当游戏程序装入内存后会自动运行，这时，按下录音机的STOP键，停止磁带转动。

## 五、打印机的使用

如果你面前的中华学习机系统配有打印机，那么掌握正确使用打印机的方法也相当重要。

### 75.如何打印西文

按前面介绍的方法将主机与打印机接好线，打开打印机电源开关，并按下打印机“ON LINE”键，看到打印机上的指示灯“READY”和“ON LINE”亮后，打开主机电源开关，引导 DOS（步骤如前面所介绍的），进入西文状态，键入：

PR#1 ↵

PRINT “ZHONG HUA XUE XI JI” ↵

执行后在打印纸上便打出“ZHONG HUA XUE XI JI”字样，在屏幕上同时也显示出“ZHONG HUA XUE XI JI”。

### 76.如何打印中文

在汉字管理程序中，打印机驱动程序的设计仅适用于九针图形打印机，其型号要求必须与 EPSON MX-80 Ⅲ 型打印机相兼容，如 FX-80 Ⅲ RX-80 Ⅱ、CP-80 Ⅲ、FX-100⁺ 和YAMATO等型号的打印机，其接口卡必须定义为 I/O 槽口的 1 号槽。

首先进入汉字系统，然后使用以下 BASIC 命令来控制和打印汉字。

（1） 设置打印方式

格式：POKE 1659，n。

其中：n＝0表示不打印，进入汉字系统后，打印方式自动置为0；n＝1～15置打印，并按系统定义的15种字型进行打印。

（2） 设置字间距

格式：POKE 1787，n。

其中：n 为所打印的ASCII字符之间的字间距，一个汉字占两个ASCII字符的位置，n 取值范围为0～255，即字间距可在0～255个点之间。进入系统后，字间距自动置为 1 。

（3） 设置行距

格式：POKE 1915，n。

其中：n 为打印行之间的点间距，n 取值范围在0～255之间，即行间距在0～255个点之间。进入系统后，字间距自动置为 1 。

（4） 设置行允许字数

格式： POKE 2043，n。

其中：n 为一行允许打印的汉字字数，一个汉字占用两个字符的位置，n 的取值范围是0～255之间。进入系统后，行允许字数自动置为40。

举例如下，

用15种字型打印“中华学习机”

程序如下：

```
10 HOME
20 FOR I=1 TO 15
30 POKE 1659, I
40 PRINT "中华学习机"
50 NEXT I
60 END
```

当上述程序键入计算机之后，键入

RUN ↵

打印机便以15种字型在打印纸上打印出：

中华学习机

## 六、磁盘和DOS系统

如果你面前的中华学习机系统除了主机、电视机（显示器）和录音机之外，还配有软盘驱动器的话，有必要认真地学习本节，掌握磁盘的使用常识和基本操作及其命令，从而可使计算机系统在更广泛的领域里得到应用。若你的系统是基本配置，即只有主机、电视机（显示器）和录音机（磁带机）的话，可以跳过本节，继续后面章节的学习。

### 77.磁盘及其使用常识

软磁盘外形象一个小唱片，是由磁性材料做成，上面可以记录大量信息，中华学习机采用 $5\frac{1}{4}$英寸的软磁盘，可以

记录 140K 字节的信息。

我们将磁盘划分成35个同心圆（图 5-2），这些肉眼看不见的同心圆，取名叫磁道，编号为0～34；每个磁道又分成16个小段，被称为扇区，编号为0～15。这样，磁盘上的信息就可供人们存取了。

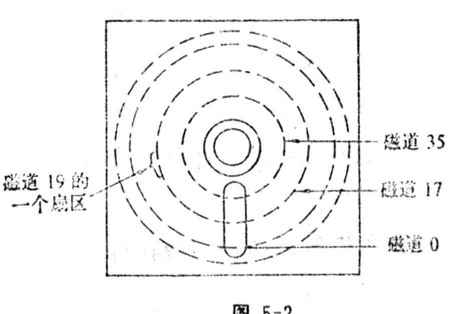


图 5-2

软磁盘被封装在保护套内。在套的上部开有一个槽形窗口，称为读写窗口，工作时驱动器的磁头，就在这个“窗口”对磁盘进行读写。

磁盘是人们保存程序和数据的工具，一定要掌握正确的使用方法，以免损坏、丢失信息。

使用时应注意：

①严禁用手去抚摸读写窗口，严禁折磁盘；

②向驱动器里插磁盘时要正面朝上（贴有标签的面），动作要轻，然后合上驱动器门柄，才能对其进行读写操作；

③当驱动器面板上的红灯亮时，表明计算机正在对磁盘进行读写工作。此时，严禁取出磁盘，待红灯熄后，方能取出和更换磁盘，否则，容易划伤磁盘，丢失信息；

④平时不用的磁盘应装入保护套中，并放在磁盘盒里。盘盒要放在清洁、干燥而没有强磁场的地方。

## 78.DOS系统

(1) 什么是DOS

DOS是英文 Disk Operation System(磁盘操作系统)的缩写。

操作系统是一个计算机系统管理程序，是人和计算机系统之间交换信息的桥梁。

这个大的计算机管理程序——操作系统设置了许多命令，控制计算机的各个部件(主机、显示器、录音机、软盘驱动器、打印机等)。人们通过操作系统可以向计算机发布命令，从而使用起来更加方便有效。

目前，大多数微型计算机将操作系统存放在磁盘里，因此称作磁盘操作系统。

中华学习机使用的磁盘操作系统是 Apple soft DOS 3.3，这是1980年8月推出的，它对前面的几个版本作了进一步的改进。

(2) 引导DOS

所谓引导DOS就是将 DOS 从磁盘装入主机内存，让计算机在DOS控制之下运行。

操作过程如下：

①首先检查主机和磁盘驱动器的连接是否正确。

②将 DOS 3.3 版本的磁盘插入驱动器中，合上驱动器盖。

③打开电视机电源，打开主机电源，这时会听到磁盘驱

动器电机转动的声音，经过5秒钟，磁盘机灯熄灭，屏幕将出现 DOS 3.3 版本号等信息说明DOS已装入机器，此过程称为冷启动。

在系统已经加电，或者系统处于 BASIC 状态下，如果需要装入 DOS 系统，此时可将 DOS 盘插入磁盘机，并键入：

]PR #6 ↵

这种在主机已加电情况下，引导操作系统的方法被称为热启动。

（3）DOS常用的命令及使用

①制作一张工作盘：在中华学习机随机提供的软盘中，有一张是 DOS 3.3 的系统主盘，盘上除了 DOS 3.3 系统程序外，还有 BASIC 解释程序等，因而占去了磁盘的大部分空间。假如我们自已要编制一些程序想存贮在磁盘上，那么所留的空间就显得太少了，另外用户程序和系统软件一起存放，稍不留神，在删除、存取等操作中，有可能会破坏系统文件，因此，建议有必要做一些工作盘，DOS 为此提供一个专用命令：

INIT

功能：格式化空白盘。

新买来的空白软盘片，上面没有记录任何信息，为了准确迅速地在磁盘上写入数据或读出数据，必须做一些标记。

用 INIT 命令可以将磁盘按磁道和扇区划分地址，以便读写磁盘文件时，能尽快地找到所需要的信息，除此之外，

在磁盘上还可建立部分磁盘管理信息，我们将这些处理过程叫做对磁盘格式化，空白盘必须经过格式化才能使用。

A.写入DOS系统程序：工作盘上写入DOS系统，每次开机，便可自动将其引导，进入机器系统，使用非常方便。INIT命令便可将DOS系统写入要制作的工作盘的0～2三个磁道上。

B.写入第一个BASIC程序：把内存的一段BASIC程序作为磁盘的第一个文件写入工作盘，以后每次引导DOS之后，便可自动执行该BASIC程序。

C.制作工作盘的操作步骤：

ⓐ将DOS 3.3系统盘插入驱动器中，做冷启动或热启动。

ⓑ启动成功后取出DOS盘，插入一空白盘片。

ⓒ键入NEW，也就是在键盘上键入N、E、W三个字母键并按回车键，便将机器中的程序清除掉。然后建立一个BASIC程序，我们暂时取名为“AUTO”，下面是此程序：功能是在屏幕上显示工作盘主人姓名、建盘日期。

```
10 PRINT "LI MING 1987.10"
20 END
```

在键入此程序时要注意有空格的地方要打入空格，每一行输完后要按回车键。

此程序建立完毕。

ⓓ键入初始化命令

```
] INIT AUTO
```

这时机器便开始对空白盘进行格式化了，大约30秒可以完成。等驱动器上灯灭之后，屏幕上出现提示符和光标，说明完成了对空白盘片的初始化操作。

经过此初始化操作的空白盘片都存有 DOS 操作系统程序，都可以作为操作系统的引导盘进行启动。

现在我们关掉主机电源数秒钟，不要将新初始化的工作盘片取出，然后打开主机电源，屏幕上便出现如下信息：

LI MING 1987.10

]

这就表明我们新建立的工作盘可以开始使用。

②列文件目录：我们在使用软盘过程中，为了存取文件方便，希望了解软盘片已经存放有哪些文件，DOS 为我们提供了此功能，通过列磁盘目录文件命令—CATALOG 可以完成这项工作。

现在我们键入命令：

CATALOG ↵

屏幕上将出现如下信息：

DISK VOLUME 254

A 002 AUTO

如果换上DOS 3.3系统主盘，再打入CATALOG命令，屏幕上将出现如下信息：

DISK VOLUME 254

*A 008 HELLO

*I 018 ANIMALS

```
 T 003 APPLE PROMS
*I 006 APPLESOFT
*I 026 APPLEVISION
*I 017 BIORHYTHM
*B 010 BOOT13
*A 006 BRIAN'S THEME
*B 003 CHAIN
*I 009 COLOR DEMO
*A 009 COLOR DEMOSOFT
*I 009 COPY
*B 003 COPY. OBJ0
*A 009 COPYA
*A 010 EXEC DEMO
*B 020 FID
*B 050 FPBASIC
*B 050 INTBASIC
```

下面我们看一看目录中各部分所代表的含意：

目录上方所显示"DISK VOLUME 254"表示当前磁盘的卷号是254（"卷号"是用来区分各个不同的磁盘的）。我们在用 INIT 进行初始化时可以给磁盘编入不同的"卷号"，例如键入：

] INIT AUTO 120 ↵

便可以建立一张工作盘，它的卷号是 120，以后每次列文件目录时，便可显示出此卷号。若在初始此时未给出卷

号，DOS自动将卷号写成 254。

目录一般分为四部分显示：

第一部分给出文件加锁的情况，凡有星号(*)的，表示该文件是加了锁的，即 DOS 系统对该文件进行写保护，只允许用户对该文件执行读操作，而不能对其执行改名、删除和写操作，否则，系统会给出：FILE LOCKED 信息。

第二部分给出文件类型：

A 表示APPELSOFT程序文件；

I 表示整数BASIC程序文件；

B 表示 2 进制代码文件；

T 表示文本文件；

R 表示由EDASM汇编程序所产生的可浮动目标文件；

L 表示LISA汇编程序所生成的文件。

第三部分是三位十进制数字，表示文件在磁盘中所占用的扇区数目。

一张磁盘共有35×16＝560个扇区，除去DOS 文件管理信息和文件目录所占去的 4 个磁通，还有 496 个扇区用来存贮程序和数据。

第四部分是文件名，它由一个以字母开头，长度不超过30个字符的字符串组成。

由于文件目录区的限制，一张磁盘最多可以存贮 105 个文件。

使用 CATALOG 命令，将首先给出磁盘上前18个文件的目录，假若你没有看到光标，表示磁盘上的文件不止 18

个 按键盘上任一键，屏幕上会接着显示，也可以用CTRL—S 键暂停列目录，再按任一键使其接着显示，

③保存程序文件：当我们在主机上编制了一个 BASIC 程序，如果想将它保存起来以后再用的话，DOS 系统为我们提供了存盘命令：SAVE，利用这个命令便会将编制好的程序保存在软磁盘上，随时可以调出运行。

例如：我们在主机上打入以下程序：

```
10 HOME
20 PRINT "您好，欢迎您使用中华学习机"
30 END
```

这是个BASIC程序，它的作用是首先将屏幕清除干净，然后显示"您好，欢迎您使用中华学习机"。当我们检查完程序认为正确，打入RUN ↵将此程序运行之后，便可以保存在工作盘上，我们把这个程序取名为 AA，打入以下命令：

```
]SAVE AA
```

执行这条命令，BASIC程序"AA"便存在磁盘上。

为了验证一下，我们打入列磁盘目录命令：

```
]CATALOG
DISK VOLUME 254
A 002 AUTO
A 002 AA
```

说明我们建立的文件 AA，已经保存在磁盘上了。

使用 SAVE 命令时要注意，保存程序的软盘片不能贴写保护，也不能是没有经过初始化的空盘片，另外，一张盘

上的文件不能重名。

④装入程序文件：当我们需要运行磁盘上某一个程序文件时必须首先将此程序从磁盘上调入内存，这要用到DOS命令LOAD。

例如，我们想把工作盘上的BASIC程序AA装入主机内存，可键入以下命令：

] LOAD AA

当屏幕上再次出现光标时，磁盘上的AA程序已经被装入主机内存，我们可以列出程序（键入LIST ↲），也可以将它运行(RUN ↲)。

在用LOAD命令装入程序时，主机内原有的BASIC程序将被清除掉，由新装入的程序来代替，而软磁盘上的程序文件不受影响，也就是说，当我们将AA调入内存后，磁盘仍然保留着程序AA。

⑤文件的删除、加锁、解锁：如果有的文件没有保存价值了，可利用DOS命令DELETE将其删除，使软盘上仅保留下有用的信息，以便腾出更多的空间来存放新的程序和数据。例如，我们想把刚才建在工作盘上的程序AA删掉，可以打入以下命令：

] DELETE AA

便可以将AA从磁盘上删除掉并不能再恢复。因此，在运行此操作时要十分小心，只有证实此文件确实无用时，再进行删除，为了防止误操作将盘上重要文件删除掉，我们可以将这些文件用LOCK命令加锁，可键入：

LOCK AA ↲

当我们再用 DELETE AA 命令来删除时，系统将给出“FILE LOCKED”（文件已加锁）的信息，若决定要删除加锁文件了，可以用解锁命令，将锁打开。键入：

UNLOCK AA ↲

然后再进行删除等操作。

⑥更改文件名：用RENAME命令，可以将磁盘上的文件名进行更换，例如要将AA文件改名为BB，可打入以下命令：

] RENAME AA，BB

执行此命令之后，我们再用 CATALOG 命令列一下磁盘目录：

] CATALOG

DISK VOLUME 254

A 002 AUTO

A 002 BB

这说明程序文件AA已经换名为BB，值得注意的是DOS不检查新名子是否与磁盘上某一个文件同名，为了避免造成混乱，换名时一定要慎重。

以上我们介绍了一些常用的 DOS 命令，掌握了这些命令，便学会了磁盘机的基本操作。除此之外，DOS 系统还提供了其它一些命令，列于书后附录中，供大家使用时参考。

（4）DOS系统的实用程序

以上介绍的命令都包含在DOS系统之中，除此之外，DOS 3.3 操作系统还提供了一些实用程序，使用这些程序能够建立主系统盘，进行软磁盘的复制，运行整数BASIC等。这些程序都存在DOS 3.3系统主盘里，使用时将程序调出。

下面列出DOS 3.3系统主盘目录及注释：

DISK VOLUME 254

| | |
|---|---|
| *A 008 HELLO | 系统启动后执行的第一个程序 |
| *I 018 ANIMALS | 猜字游戏程序 |
| T 003 APPLE PROMS | 由RANDOM程序建立的随机文件 |
| *I 006 APPLESOFT | 装入APPLESOFT解释程序的实用程序 |
| *I 026 APPLEVISION | 动画图形表演程序 |
| *I 017 BIORHYTHM | 计算人体生物曲线的游戏程序 |
| *B 010 BOOT13 | 13扇区软盘的引导程序 |
| *A 006 BRIAN'S THEME | 高分辨率作图示范程序 |
| *B 003 CHAIN | APPLESOFT 连接程序 |
| *I 009 COLOR DEMO | 彩色示范程序 |
| *A 009 COLOR DEMOSOFT | 彩色示范程序 |

| | |
|---|---|
| *I 009 COPY | 软盘复制程序 |
| *B 003 COPY.OBJ0 | 软盘复制程序所需用到的RWTS |
| *A 009 COPYA | 软盘复制程序 |
| *A 010 EXEC DEMO | EXEC命令的示范程序 |
| *B 020 FID | 软盘管理程序 |
| *B 050 FPBASIC | APPLESOFT 解释程序 |
| *B 050 INTBASIC | 整数 BASIC 解释程序由HELLO程序装入 |
| *A 003 MAKE TEXT | 建立文本文件的示范程序 |
| *B 009 MASTER CREATE | 建立主系统盘的使用程序 |
| *B 027 MUFFIN | 将13扇区的文件转换成16扇区文件的实用程序 |
| *A 051 PHONE LIST | 存贮电话号码的应用程序 |
| A 010 RANDOM | 使用随机文件的库存管理示范程序 |
| *A 013 RENUMBER | 对 APPLESOFT 程序可重新编行号的使用程序 |
| *A 039 RENUMBER | RENUMBER 实用程 |

INSTRUCTIONS　　　　序的使用介绍

＊A 003 RETRIEVE TEXT 取文本文件的实用程序

对一些非常有用的使用程序，我们将在附录里给出其使用方法。

(5) 在BASIC程序中使用DOS命令

前面介绍的DOS 命令，都是由键盘输入直接执行的。另外在 BASIC 程序中，DOS 命令也可以作为一条特殊的BASIC语句被调用。

在BASIC程序中使用 DOS 命令，必须将 DOS 命令写在PRINT语句中，并且必须在前面加上CTRL—D

调用DOS命令的语句格式：

PRINT "CTRL—D DOS 命令和参考"

其中CTRL—D的键入方法是，按住CTRL 键，再敲D键，同时必须保证CTRL—D前是回车字符。

另外，也可以用CHR＄(4)来代替 CTRL—D，格式如下：

PRINT CHR＄(4)； "DOS命令和参数"

例如下面一段程序将显示磁盘目录：

```
10 HOME
20 PRINT CHR$(4); "CATALOG"
30 END
```

通过这一章的学习，我们基本上可以掌握磁盘操作了，但是，如想得心应手地运用 DOS 命令，还需要反复实践才行。

# 七、中华学习机的维护与保养

## 79.微机发生故障的一般原因

微型计算机具有工作可靠、速度快、使用方便等优点。但是，它也和其它机器一样，会出现故障，造成故障的一般原因，大致有以下几种：

（1）电网电压不稳定；突然掉电和加电；瞬态脉冲电压过高，都有可能使微机造成损坏。

电压不稳定一般是电压不足，当电压在 170V 左右时，电源内的调压器就不能供给足够的电压，从而使数据显示混乱。有时，还会造成系统工作间断。

突然掉电与加电时，会有高频脉冲产生，这些尖峰信号偶然也会越过滤波保护进入逻辑电路，一般情况是抹掉或改变数据，但有时也会损坏电路器件。

瞬态高频脉冲电压干扰是与回路中负载情况有关，也与外界磁场有关。如马达、电焊机的启停等，会突然产生一个维持几周或几十周的高幅度变化的电压现象，这种现象的发生对机器的危害最大。

（2）工作环境差，灰尘过多，会使机器出错划盘。温度过高，元件内部过热，会使元件过早老化而坏掉。

灰尘会被计算机和显示器中的静电荷吸附。当灰尘过多时，会阻碍元器件散热，机内温度就会比正常情况要高。因

此，会使芯片及其它组件坏得更快。尤其是磁盘驱动器因为灰尘而造成的故障最多，这是因为在磁盘驱动器内部，是依靠磁头来进行读写的，磁头和磁盘之间的距离非常小，当磁头悬浮在磁盘表面上时，灰尘和杂质会引起许多故障，外来杂质微粒如烟尘、纤维，都可能引起磁盘驱动器突然的故障。

当机器内部的适配器接口板增多时，热量就不会即时排除，造成机器内部温度过高，这样就会造成电子元器件过早老化、接点处断路，当温度过高时，就会出现丢失数据等现象。

(3) 人为事故。如摔坏、插错电源、插错插件位置，不按操作规程办事，等等。也有的使用不当，如将打印机装错纸，打印头卡纸，使打印头损坏，或烧坏电机，这些也是造成微机出现故障的因素。

当然除了以上原因之外，机器本身工作不稳定、组件失效也是造成机器故障的原因。但这也与使用条件有密切关系。因而要想减少事故，延长机器寿命，就要清除一切对机器不利因素，做好日常维护工作。

## 80.中华学习机的维护与保养

(1) 建立一个良好的工作环境

工作环境对中华学习机的运行和使用寿命有很大影响。

中华学习机的工作环境应该是这样的：温度在10～40℃，太冷太热都会使机器工作不稳定甚至无法工作。相对湿度一

般在20～80％，最好在35～65％之间，应尽量避免日光照射。这是因为显示器的工作主要是靠荧光粉的作用，日光的照射将会使荧光粉脱落，从而造成显示器不能正常工作。因此，最好做一个罩子将机器罩住，既可以避免日照，又可以防止灰尘落入机器。当机器内积有灰尘时，要及时清除掉，可以用酒精棉球来进行擦试。

（2）要有一个稳定的电源

中华学习机和其它微机一样对电压的波动非常敏感，要想使设备的故障减少，就必须有一个稳定的电源。

对于电压不稳定的地方要使用稳压电源；对于经常停电又突然加电的地方，可以使用不间断电源。这种电源是在外界供电停止后，自动将内部的蓄电池中的直流电换成交流电，当外界供电时，它又自动停止工作，把外电源自动接入。

（3）磁盘和磁盘驱动器的保护

①磁盘的保护：

A．不要弯曲、折叠磁盘，不要将磁盘靠近强磁场。

B．不能用手摸磁盘上的读写窗口，应当保持清洁，防止灰尘落入。

C．注意磁盘的工作环境，温度、湿度、太冷、太热都会使磁盘变形。

D．不要在已贴上磁盘的标签上写字，因为圆珠笔及铅笔会在盘表面留下凹痕。应先写完标签，再将标签贴在盘套上。

②软盘驱动器的日常保养：软盘驱动器使用率最高，也最容易出故障。因此，要维护好机器，首先要维护好软盘驱动器。软盘驱动器之大忌是灰尘，多数故障的出现是划盘，即把磁盘表面的磁涂层磨掉了，从而不能正确地记录数据而出现错误。因而，清洗磁头是磁盘驱动器日常保养的重要工作。其方法如下：

A.用清洗盘擦磁头：市场上有一种清洗盘，可以用它来清洗磁头。它是一种特制的软盘，灌入清洁液，放入驱动器中，使它转动后，则灰尘和磁头上的污垢被清洗盘所吸收。

B.用清洁液擦洗磁头：用一个棉花棒沾上酒精或商店里买来的擦录音机磁头用的清洁液来擦洗磁头。用棉花棒擦磁头一定要小心，不要把磁头碰坏。

以上介绍了中华学习机最基本的维护和保养知识，只要坚持日常维护，又能按照正确的操作规程使用，你的中华学习机就一定会延长寿命，经常处于良好的工作状态。

# 附录A 中华学习机的作图功能(BASIC)

## 一、中华学习机的显示方式

中华学习机共有五种显示方式，主要分为两大类：一类为文本显示方式，一类为图形显示方式。图形显示方式又分为低分辨率图形和高分辨率图形。混合方式是字符和图形同时在屏幕上显示，无论在低分辨率和高分辨率方式下，屏幕下方都留有四行的文本显示区。

### 1.文本显示

全屏幕24行，每行40个字符，共计 960 个字。显示方式采用①正常方式——黑底白字；②反向方式——白底黑字；③闪烁方式——正常方式与反向方式交替。

### 2.低分辨率图形显示

低分辨率图形显示是由颜色块组成的，每一行有40块颜色块，但颜色块的行数比文本显示的字符行数多一倍。若把低分辨率图形的紧挨着的两行看作一行，每一块颜色块由上、下两块组成，那么一整块颜色块就与文本显示时一个字符的大小一样了。在低分辨率状态下，可用40×40，即1600块颜色块显示图形，用最下面的四行显示文本。

### 3.高分辨率图形显示

高分辨率图形显示280×192个彩色点的点阵，高分辨率图形显示数据存贮在两个八K的存贮区中，这两个存贮器分别叫做显示页面1和显示页面2。

在显示高分辨率图形时，屏幕下面的四行显示文本。

以下几个命令用于低分辨率彩色作图。

(1) GR

功能：该语句的作用是将屏幕置成低分辨率作图状态，这时，屏幕上的图形显示窗口为40×40，窗口下留4行供文本使用。GR 命令自动将屏幕清为黑色，光标移到文本行的开头，因此，GR 命令有清除低分辨率图像画面的作用，但同时原图像区的图像也被破坏。

(2) COLOR

格式：COLOR=〈算术表达式〉

功能：在低分辨率作图状态下选择画面的颜色。如果算术表达式是一个实数，它会被转换为一整数。算术表达式的取值范围是0～255，但被处理为0～15。0～15的数与16种颜色对应如下：

| | | | |
|---|---|---|---|
| 0 | 黑色 | 8 | 棕色 |
| 1 | 红色 | 9 | 橙色 |
| 2 | 深蓝色 | 10 | 灰色 |
| 3 | 紫色 | 11 | 粉红色 |
| 4 | 深绿色 | 12 | 浅绿色 |
| 5 | 灰色 | 13 | 黄色 |

6　中蓝色　　　　　14　绿蓝色

7　浅蓝色　　　　　15　白色

GR 命令将 COLOR 置 0，即为黑色。

(3) PLOT

格式：PLOT(算术表达式 1)，(算术表达式 2)

功能：在低分辨率作图状态中，将一个点（即一块颜色块）放在坐标为（算术表达式 1，算术表达式 2）处。点的颜色由最近执行的COLOR语句决定。

由于低分辨率作图状态下，屏幕的图形显示窗口为40×40，所以算术表达式 1 和算术表达式 2 的取值范围为 0～39。

(4) HLIN

格式：HLIN(算术表达式 1)，(算术表达式 2)，
　　　AT(算术表达式 3)

功能：在低分辨率作图状态下，画一条从点（算术表达式 1，算术表达式 3）到点(算术表达式 2，算术表达式 3）的水平直线，线的颜色由最近执行的 COLOR 语句决定。

算术表达式 1，算术表达式 2，算术表达式 3 的取值范围是0～39。

(5) VLIN

格式：VLIN（算术表达式 1），（算术表达式 2），
　　　AT（算术表达式 3）

功能：在低分辨率作图状态下，画一条从点（算术表达式 3，算术表达式 1)到点(算术表达式 3，算术表达式 2）

的垂直直线，线的颜色由最近执行的COLOR语句决定。

(6) TEXT

格式：TEXT

功能：作用是把屏幕从低分辨率作图状态或两种高分辨率作图状态之一返回全屏幕文本状态（每行40个字符，共24行）。

下面给出几个简单的低分辨率状态下作图的例子。

[例 1] 在屏幕上显示16种颜色。

```
10 GR
20 HOME
30 FOR I=1 TO 31
35 COLOR=I/2
40 VLIN 0, 39 AT I
50 NEXT I
60 END
```

[例 2] 在屏幕上显示不断变换的图案花，

```
10 GR
20 FOR A=3 TO 5
30 FOR B=1 TO 19
40 FOR C=0 TO 19
50 S=B+C
60 COLOR=C+3/(B+3)+B*A/12
70 PLOT B, S
```

```
80 PLOT S, B
90 PLOT 40-B, 40-S
100 PLOT 40-S, 40-B
110 PLOT S, 40-B
120 PLOT 40-B, S
130 PLOT B, 40-S
140 PLOT 40-S, B
150 NEXT C
160 NEXT B
170 NEXT A
180 END
```

## 二、高分辨率作图

高分辨率作图共有以下几条命令：

(1) HGR

格式：HGR

功能：将屏幕置成高分辨率作图状态，屏幕清为黑色，显示窗口的总网格数为280（X方向）×160（Y方向），窗口下留四行供文本使用。这个命令显示的是高分辨率作图状态下存贮器(8K～16K)的页面1。

(2) HGR2

格式：HGR2

功能：这个命令置全屏幕为高分辨率280（X方向）×192（Y方向）作图状态，设有文本说明行，使用的是存贮

器(16K～24K)的页面2。

(3) HCOLOR

格式：HCOLOR＝（算术表达式）

功能：由算术表达式的值决定高分辨率作图的颜色。算术表达式的取值范围是0～7，对应颜色为：

| | | | |
|---|---|---|---|
| 0 | 黑色1 | 4 | 黑色2 |
| 1 | 绿色 | 5 | 红色 |
| 2 | 紫色 | 6 | 黄色 |
| 3 | 白色1 | 7 | 白色2 |

由 HCOLOR＝（算术表达式）命令所画的点的颜色还取决于该点在屏幕上的位置。例如HCOLOR＝1，该点处在偶数列(0，2，4，…，278)则显示紫色，处在奇数列(0，1，3，…，279)则显示绿色。HGR和HGR2命令都将HCOLOR置为0，即显示黑色。

(4) HPLOT

格式：①HPLOT（算术表达式1），(算术表达式2）

②HPLOT TO（算术表达式3），（算术表达式4）

③HPLOT（算术表达式1），(算术表达式2）
TO（算术表达式3），（算术表达式4）
TO（算术表达式5），（算术表达式6）

功能：

①该命令在X＝（算术表达式1）， Y＝（算术表达式2）（X， Y）上画一个高分辨率的点，颜色由最近执行的

HCOLOR语句决定。

②该命令从上次画的点到点（算术表达式3，算术表达式4）画一条直线。如果画完上一点后，HCOLOR 的值已改变，但这条线的颜色仍由上次画的点的颜色决定；若前面没有画过一个点，那么这条命令不能画一条线。

③该命令从点（算术表达式1，算术表达式 2 ）到点（算术表达式3，算术表达式4）画一条直线，颜色由最近执行的HCOLOR 语句决定，如果对这个命令进行扩展，所画线几乎可以不受限制地延伸。所受限制只有两点：a.屏幕限制；b.每个命令最多239个字符。

HPLOT命令必须在HGR和HGR2被执行后方可执行。X方向的横坐标取值范围是0～279，Y 方向的坐标取值范围是0～191。

要从高分辨率作图状态回到文本状态，只要打入命令TEXT即可。

下面给出一段程序，运行后将在屏幕上画出一条抛物线：

```
10 HGR
20 A=-11
30 B=11
40 C=0.1
50 X1=A*8+140
60 Y1=158-(A↑2)*1.2
70 HCOLOR=3
```

```
80 HPLOT 0, 158 TO 279, 158
90 HPLOT 140, 160 TO 140, 0
100 FOR X=A TO B STEP C
110 X2=140+8*X
120 Y2=158-(X∧2)*1.2
130 HPLOT X1, Y1 TO X2, Y2
140 X1=X2
150 Y1=Y2
160 NEXT X
170 END
```

利用高分辨率可以作出许多图形，关键在于计算点的坐标，大家可以自己下去多实践。

中华学习机除了上面介绍的几种作图外，还有其它方法，这里就不作介绍了。

# 附录B　中华学习机上的音响(BASIC)

在中华学习机的机盒内，有一只 0.25W8Ω 的扬声器，在软开关的控制下，扬声器可以发出各种响声。

那么怎样才能让扬声器发出的声音变成优美的乐曲呢？这就要取决于音调和发声的长短。

声音的音调取决于信号的频率，下面的表格列出了音阶和对应的频率（频率用 $x$ 值表示）：

| x值 | 255 | 228 | 205 | 192 | 171 | 152 | 140 | 128 | 114 | 102 | 95 | 84 | 75 | 68 | 62 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 音阶 | $\underset{\cdot}{5}$ | $\underset{\cdot}{6}$ | $\underset{\cdot}{7}$ | 1 | 2 | 3 | 4 | 5 | 6 | 7 | $\dot{1}$ | $\dot{2}$ | $\dot{3}$ | $\dot{4}$ | $\dot{5}$ |

发出声响的长短取决于信号维持时间的长短，下表列出了发声时间与对应的音长（Y值代表发声时间）：

| $y$值 | 30 | 70 | 110 | 160 | 255 |
|---|---|---|---|---|---|
| 音长 | 1/4拍 | 1/2拍 | 1 拍 | 2 拍 | 4 拍 |

当我们规定好音阶和音长之后，用下面的语句便可以使扬声器奏出乐曲来。

语句 MUSIC

格式：MUSIC X，Y

功能：定义一个音阶和音长，并使扬声器发声。

〔例〕 下面一段程序可以使扬声器奏出音节旋律。

```
10 FOR I=1 TO 15
20 READ X，Y
30 MUSIC X，Y
40 NEXT I
50 DATA 255，160， 228， 160， 205， 160， 192，
   160，171，160，152，160，140，160，128，160，
   114，160，102， 160， 95， 160， 84， 160， 75，
   160，68，160，62，160
```

除了用 MUSIC 语句可以让中华学习机演奏乐曲之外，还可以用PEEK函数（读内存某一单元内容）去读与扬声器有联系的-16336号存贮单元，也能够达到让中华学习机演奏音乐的目的。

# 附录C　出错信息表

当一个错误发生后，程序不能继续执行，系统给出错误信息，我们可以根据出错信息，找出程序中的错误，修正之后，继续运行。

出错信息的格式如下：

立即执行方式的语句：?XX ERROR

间接执行方式的语句：?XX ERROR IN YY

其中：XX：表示错误码，也就是错误信息；

　　　YY：产生错误的语句行号。

以下列出可能产生的出错信息及其含义：

CAN'T CONTINUE

执行一个不存在的或者已经出错的程序，或者想执行某个已经误删了一些行的程序。

DIVISION BY ZERO

除数为零。

ILLEGAL DIRECT

在立即执行方式中，使用了INPUT、DEFFN，GET或DATA等语句。

ILLEGAL QUANTITY

函数的自变量违反了数学的定义或超出了系统允许的取值范围，主要有以下几种：

①负数为数组下标变量（例如：A(−1)=3)。

②用LOG函数时，自变量为负数或零。

③用SQR函数时，自变量为负数。

④A∧B中B不是整数时，A为负数。

⑤在MID $，LEFT $，RIGHT $，WAIT，PEEK，POKE，TAB，SPC，ON… GOTO 或作图的任一函数中使用了不恰当的自变量。

NEXT WITHOUT FOR

有NEXT语句而无与之对应的FOR语句，或NEXT语句中的变量与配对的FOR语句中的变量不一致。

OUT OF DATA

执行一个READ语句时，程序中的所有 DATA 语句早已被读完，而程序试图要读更多的数据。

OUT OF MEMORY

程序太长，变量太多，FOR循环嵌套多于10层，GOSUB多于24层，表达式太复杂，括号嵌套超过36层，置LOMEM或HIMEM时太高或太低都属于这种情况。

FORMULA TOO COMPLEX

执行到多于两条 IF “XX” THEN形式的语句。

OVERFLOW

计算的结果太大，超出机器系统的处理范围，如果产生下溢，屏幕上只出现零，而没有任何出错信息。

REDIM'D ARRAY

对一个数组两次使用DIM语句时产生此错误。这个错误经常发生的情况是：如果直接使用像 A(I) =5这样的语句，则BASIC 自动定义其标量长度为10，但在程序中又跟有如DIM A(100)的语句。

RETURN WITHOUT GOSUB

在程序中没有相应的GOSUB 语句，而出现 RETURN 语句。

STRING TOO LONG

一个字符串（可能由几个字符串连接而成的）长度超过了 255 个字符。

BAD SUBSCRIPT

引用了不在数组维数定义范围之内的数组元素。如当A已经用DIM A(3，2) 定义了，却又出现了A(2，2，2)=B的语句。

SYNTAX ERROR

在语句中出现了句法错误，比如使用不正确的标点符号，或遗漏括号等等。

TYPE MISMATCH

赋值语句中等号两边的类型不匹配，如左边用数值量，而右边却是字符串。另一种情况是一个函数的参数类型是一个串，但给出的却是一个数值型数据，或者相反。

UNDEF'D STATEMENT

使GOTO、GOSUB或THEN语句转向程序中不存在的

语句标号。

UNDEF'D FUNCTION

使用了用户未曾定义过的函数。

# 附录D DOS3.3磁盘操作系统命令及使用

## 一、DOS命令格式

DOS命令由命令保留字和命令参数二部分组成。

命令保留字代表了 DOS 操作系统所提供的命令，命令参数则是提供 DOS 命令执行时所需要的辅助信息，对可有可无的参数，用方括号[ ]形式给出。

DOS命令格式如下：

COMMAND f [, Vn]

其中，COMMAND：为命令保留字；

f：文件名，是以字母打头的字符串（其中不能含有逗号）；文件名的有效长度最多可达30个字符；

Vn：为磁盘的卷号，一张磁盘只能有一个卷号，*n*的值可以是1～254之间的整数。

方括号表示此项为可选项。

## 二、DOS命令的执行方式

在引导DOS操作系统之后，调用 DOS 命令的方法有两种：①立即方式；②程序方式。

**1.立即执行方式**

在BASIC提示符下，直接打入DOS命令，DOS将立即执行该命令，这种方式称为立即执行方式。

可以立即执行的 DOS 命令有：BLOAD、BRUN、BSAVE、CATALOG、CLOSE、DELETE、EXEC、FP、INIT、IN#、INT、LOAD；LOCK；MAXFILES；MON；NOMON；PR#；RENAME；RUN；SAVE；UNLOCK VERIFY

**2.程序调用方式**

在BASIC 程序中，DOS 操作系统命令均可作为一条特殊的BASIC 语句被调用，它与一般的 BASIC 语句不同的是：DOS 操作命令必须通过一条特殊的 PRINT 语句来执行。这条PRINT语句将DOS 命令及其命令参数作为一输出行进行输出，而在这输出行的第一个字符是CTRL-D。

CTRL-D是DOS命令的引导标记，当BASIC解释程序执行PRINT语句时，遇到第一个输出字符是 CTRL-D，则作为DOS命令去处理。

**3.调用DOS命令的语句格式**

PRINT “CTRL-D DOS命令和命令参数”

或 PRINT CHR $(4)；“DOS命令和命令参数”

使用该语句应注意以下几点：

(1) 在一个PRINT语句中，只能含有一条DOS 命令，而在这个程序行中必须以CTRL—D 字符或 CHR $(4) 开头，以回车(RETURN)或回车字符(CTRL—M) 表示DOS

命令的结束。

(2) 在此语句之前的最近一条 PRINT 语句，必须以回车符结束。

仅能以程序调用方式执行的DOS命令有

APPEND；CLOSE；OPEN；POSITION；

READ；WRITE；

## 三、初始化命令

格式：INIT f [，Vn]

功能：初始化软盘，并把已建立在内存中的 BASIC 程序作为 DOS 操作系统引导后，装入并运行的第一个程序存入软盘，冠以文件名 f 。

## 四、管理磁盘文件的DOS命令

### 1.CATALOG 命令

格式：CATALOG

功能：显示盘片的文件目录，目录中包括：磁盘的卷号，文件的名称，文件所占用盘片的扇区号，文件的类型和文件的封锁标志等。

文件的类型显示如下：

A：表示Applesoft BASIC程序

I：表示整数BASIC程序

B：表示二进制文件

T：表示文本文件

**2.LOCK命令**

格式：LOCK f [，Vn]

功能：该命令将对指定的文件置封锁状态，也就是写保护状态，使得该文件不会因意外的写、修改、删除、命名等操作而遭受破坏。对于被封锁的文件，在列磁盘文件目录时，在其最前面将有一星号(*)。

**3.UNLOCK 命令**

格式：UNLOCK f [，Vn]

功能：解除对指定文件的封锁，使得此文件可以被删除，重新命名、修改或重写，解除文件封锁后，封锁标志将消失。

**4.RENAME命令**

格式：RENAME $f_1$，$f_2$ [，Vn]

功能：修改文件名，用文件名$f_2$代替文件名$f_1$，改名后文件内容不受影响。

这里$f_1$是一个已存在软盘上的文件名，$f_2$则是软盘上没有的新名；$f_1$文件不被加锁。

例如，RENAME AAA，BBB

执行此命令后磁盘上原有的文件AAA，将改名为BBB。

**5.VERIFY命令**

格式：VERIFY $f_1$ [，Vn]

功能：通过读操作，对指定文件f进行检查，以确定该文件在软盘上的信息是否正确。如果被校验的文件正确，将得不到任何回答信息，否则，将给出210 ERROR错误信息

（意思是输入/输出错误）。

### 6.DELETE命令

格式：DELETE f [，Vn]

功能：从软盘上删除所指定的文件，并且该文件不被加锁。

如果要删除的文件不存在，将出现 FILE NOT FOUND；如果该文件被加锁，将得到信息 FILE LOCKED。

## 五、与BASIC程序有关的DOS命令

### 1.LOAD命令

格式：LOAD f [，Vn]

功能：从软盘上装入一个BASIC程序。在装入BASIC程序时，盘上的文件保持不变，原先已在内存的 BASIC 程序将被清除。

### 2.SAVE命令

格式：SAVE f [，Vn]

功能：将已建立在主机内存中的 BASIC 程序副本传送到软盘上。如果该文件名在软盘上已存在，而且文件类型与系统的当前状态相吻合，则用当前已在内存中的 BASIC 程序替代原有文件。因此，在用SAVE命令向软盘上传送文件时，要避免重名。

### 3.RUN命令

格式：RUN f [，Vn]

功能：把软盘上的某一 BASIC 程序 f 装入到内存并运行。它的执行步骤是：①执行DOS 的 LOAD命令；②执行BASIC的RUN命令。

## 六、语言之间进行转换的DOS命令

DOS 3.3 操作系统提供了 APPLESOFT 解释程序和整数BASIC解释程序二种语言相互切换的命令。

### 1.INT命令

格式：INT

功能：使系统进入整数 BASIC 状态，并清除内存中现有程序，恢复程序有关指针。

### 2.FP命令

格式：FP

功能：APPLESOFT程序又称为浮点BASIC程序，该命令将使得系统进入APPLESOFT 状态，并清除内存中现有的BASIC程序，恢复各有关指针。

## 七、对文本文件进行操作的DOS命令

文本文件是用来存放数据信息的文件，其数据信息包括运算数据、计算结果、信件的副本、帐单和表格等。

文本文件按其存取方式可以分成顺序文件和随机文件。这两种文件在用CATALOG 命令列磁盘目录时，均以字母T作为类型标志符。但是这两种文件的使用方法和存贮格式有所不同。下面分别介绍这两种文件的存取方式及与其对应的

DOS命令。

**1.顺序文件**

顺序文件由数据域组成，每个数据域由回车字符作为终止标志。通常由不带逗号和分号的 PRINT 语句自动生成，也可以由 CHR $(13) 字符来产生。顺序文件在软盘上是连续存放的，各个数据域之间没有间隙，其数据域的长度和文件长度可是任意的。

(1) OPEN命令

格式：OPEN f [, Vn]

功能：打开一顺序文件 f，如果该文件不存在，则在磁盘的目录上创建该文件。在对某一文件进行读操作或写操作之前，首先必须打开该文件。程序一次打开的文件个数是由 MAXFILES命令所决定，最多不得超过16个，DOS启动后将自动执行MAXFILE3 命令，此时系统允许打开的文件个数最多为 3 个。

(2) CLOSE命令

格式：CLOSE [f]

功能：关闭名为 f 的文件，当文件名省略时，将关闭所有已打开的文件。对某一文本文件在完成读 / 写操作后必须关闭，以收回分配给该文件的文件缓冲区。

(3) WRITE命令

格式：WRITE f [, Bn]

功能：对已打开的文件 f 执行写操作。参数 Bn 指出从顺序文件的第 n 个字节开始写数据，如果此参数省略，将从

第 0 个字节开始写数据。

(4) READ命令

格式：READ f [，Bn]

功能：对已打开的文件 f 执行读操作，参数 Bn 指出从顺序文件第 n 个字节开始读数据，如果此参数省略，将从第 0 个字节开始读数据。

(5) APPEND命令

格式：APPEND f [，Vn]

功能：自动打开顺序文件 f ，并把文件指针移到文件的末尾，然后利用 WRITE 命令可以在文件的末尾增加新的信息。

(6) POSITION命令

格式：POSITION f [，Rn]

功能：用来对顺序文件 f 执行定位操作，可以把文件位置指针从当前位置，移到下面第 n 个域的起始位置。参数Rn指出当前位置的第 n 个域，当参数Rn省略时，表示不移动文件的位置指针。例如，POSITION G2，R3 即将文件位置指针移至顺序文件G2的第 3 个域的起始位置。

POSITION 命令必须用在WRITE和READ命令之前，否则，使用过POSITION命令之后，还需要重新使用READ或WRITE命令才能执行读操作或写操作。

**2.随机文件**

随机文件由记录组成，以记录为单位进行数据的读操作和写操作，每个记录中可以包含若干个数据域。各个记录的

长度是固定的；一个记录的最大长度是32767个字符。

使用随机文件，可以实现快速查找，增补或修改其中任一记录中的数据，然而却不会影响其它记录中的内容。

(1) OPEN命令

格式：OPEN f，L [，Vn]

功能：打开一随机文件 f，其记录长度由参数L来决定，它是1～32767之间的数。

(2) CLOSE命令

格式：CLOSE [f]

功能：关闭名字为 f 的文件，若不给出文件名，将关闭所有打开的文件。

(3) WRITE命令

格式：WRITE f [，Rn] [，Bn]

功能：对随机文件 f 进行写操作。

Rn：指出对随机文件中执行写操作的记录号，若Rn 省略，将从记录 0 开始写。

Bn：指出从该记录第 n 个字节开始执行写操作，若 Bn 省略，将从此记录的 0 字节开始写。

(4) READ命令

格式：READ f [，Rn] [，Bn]

功能：对随机文件执行读操作。

Rn：n 为 0～32767 之间的数字。指出对随机文件中执行写操作的记录号，若Rn省略，将从记录 0 开始写。

Bn：指出从该记录第 n 个字节开始执行读操作；若 Bn

省略，将从此记录的0字节开始写。

## 八、对2进制文件进行操作的DOS命令

2进制文件存放的是主机内存中的2进制信息副本，即把主机内存中某个地址段的信息以文件的形式保存在软盘上。在文件目录中，2进制文件的类型标志是B文件。

(1) BSAVE命令

格式：BSAVE f，An，Ln [，Vn]

功能：把内存中以An为起始地址，长度为Ln的二进制文件f保存在软盘上。

An：n是0～65535之间的内存地址，指明了要存储的2进制信息在内存中的起始地址。

Ln：n是0～32767之间的数，指出要传送的内存字节数。

(2) BLOAD命令

格式：BLOAD f [，An] [，Vn]

功能：将盘上的2进制文件f装入到主机内存由参数An为首地址的区域中。

An：n是0～65535的内存地址，表示文件f存入内存的首地址；若此参数省略时，其起始地址为该文件由BSAVE写入时所对应的An值。

(3) BRUN命令

格式：BRUN f [，An] [，Vn]

功能：首先装入以f为文件名的2进制文件，然后转移

到该文件以参数An为起始地址去执行。其中参数An所给出的地址要和建立该文件时起始地址相一致。

## 九、辅助命令

(1) MON命令

格式：MON [C] [, I] [, O]

功能：在用程序方式调用 DOS 命令时，DOS命令及主机与磁盘之间的信息交换在屏幕上不表现出来，为了方便程序的调试，我们可以用MON命令来实现监视信息。

此命令有三个参数可供选择：

C：显示所有的DOS命令；

I：显示由软盘读入主机的信息；

O：显示由主机写入软盘的信息。

(2) NOMON命令

格式：NOMON [C] [, I] [, O]

功能：终止 MON 命令的执行，也就是撤除监视。其参数的含义和使用方法和MON命令完全相同。

(3) PR#

格式：PR#n

功能：将输出字符送到第 n 号槽口所连接的输出设备上。n 可以是0～7之间的整数。

对于中华学习机来说，n＝0 表示屏幕显示，n＝3表示进入汉字系统，n＝6表示从软盘上引导DOS操作系统。

如果我们在中华学习机扩充槽内接有打印机且槽口定义

为1号，要打印数据，可用PR#1来选定打印机。

(4) IN#命令

格式：IN#n

功能：从第n号槽口所连接的输入设备接受信息，其中n可以是0～7之间的整数。

对于中华学习机来说，n＝0表示从键盘接受信息。

(5) MAXFILES命令

格式：MAXFILES n

功能：指出一次能够打开的文件数，n为1～16之间的整数。

(6) EXEC命令

格式：EXEC $f_1$ [，Rn] [，Vn]

功能：指导DOS从命令文件而不是从键盘获取命令，且可以从命令文件的任何一行开始执行。

其中，Rn的n是0～32767之间的数字，指出命令文件跳过的行数，如果省略，DOS将从命令文件中第一行开始执行。

# 掌握现代智力开发工具——中华学习机

史瑞芬　岳　琨　编

责任编辑　刘　嘉

河南科学技术出版社出版

河南第一新华印刷厂印刷

河南省新华书店发行

787×1092毫米　32开本　9.125印张　175千字

1988年9月第1版　　1988年9月第1次印刷

印数 1—10000册

ISBN7-5349-0265-7/Z·266

定价：　2.50元

ISBN 7-5349-0265-7/Z·266
定价 2.50元

责任编辑／刘嘉
美术设计／胡斌

掌握现代智力开发工具——中华学习机

河南科学技术出版社